



RAPPORT DE Projet Tuteuré & Machine Learning

Soutenu le : 15 juillet 2025

**Réalisé par : Balkis Benslimane, Samar Omrani , Ahmed Makhlouf ,
Nadhir Maamar et Sarah Ghorboal**

**« Segmentation et Analyse du Comportement Client pour la
Réduction du Churn »**

Encadrant académique : Aymen Ben Brik & Heni Abidi

Année universitaire 2024/2025

Introduction Générale	6
1. 6	
Introduction	7
1.1 Contexte du projet	7
1.2 Objectifs	7
1.3 Méthodologie adoptée	8
2. 9	
Introduction	9
2.1 Nettoyage des données	9
2.2 Traitement des valeurs manquantes	10
2.3 Encodage des variables catégorielles	11
2.4 Mise à l'échelle des variables (si nécessaire)	11
2.5 Sauvegarde du dataset traité	12
Conclusion	12
3. 14	
Introduction	13
3.1 Extraction et nettoyage des données	13
3.2 Transformation des données avec Talend	13
3.2.1 Exemple de job Talend : intégration des données clients	13
3.2.2 Construction de la table de faits à l'aide de Talend	15
3.3 Chargement dans PostgreSQL	16
Conclusion	17
4. Création d'un tableau de bord avec Power BI	18
Introduction	18
4.1 Connexion à la base de données PostgreSQL	18
4.2 Tableau de bord	19
4.2.1 Mesures Clés Utilisées dans le Tableau de Bord	20
4.2.2 Méthodologie de Calcul du Taux de Churn à partir d'une Colonne Texte	20
4.2.3 Visualisations	21
4.2.4 Filtres Interactifs	22
Conclusion	23
5. 26	
Introduction	24
5.1 Analyse exploratoire et préparation des données	24
5.2 Analyse statistique univariée	25
5.3 Analyse bivariée	27

5.4 Ingénierie des données	29
5.5 Modélisation et comparaison des algorithmes	29
5.6 Synthèse des performances	33
Conclusion	34
6.	39
Introduction	35
6.1 Préparation des données	35
6.1.1 Nettoyage et traitement	35
6.1.2 Agrégation temporelle	35
6.2 Analyse exploratoire de la série temporelle	35
6.2.1 Composantes de la série	35
6.2.2 Stationnarité de la série	36
6.3 Modélisation prédictive	37
6.3.1 Modèle ARIMA (1,1,1)	37
6.3.2 Modèle SARIMA (3,1,0) (1,1,0) [7]	37
6.3.3 Modèle XGBoost	38
6.3.4 Modèle LSTM	39
6.4 Évaluation et comparaison des modèles	39
Conclusion	40
7. Déploiement	46
Introduction	46
7.1 Gestion des Environnements Conda	46
7.1.1 Gestion des requirements	46
7.1.2 Lancement Réussi de l'Application	47
7.2 Authentification au dashboard	47
7.2.1 Sécurité des Données	48
7.2.2 Personnalisation Utilisateur	48
7.3 Power BI	48
7.4 Prédiction	48
7.5 Prévision	50
Conclusion	52
Conclusion Générale	42

Introduction Générale

Dans un marché des télécommunications de plus en plus concurrentiel, la fidélisation de la clientèle constitue un enjeu stratégique majeur. La résiliation des contrats, ou *churn*, représente une perte significative pour les opérateurs, tant en termes de revenus que d'image. Comprendre les comportements clients, anticiper les départs et identifier les leviers d'action sont devenus essentiels pour maintenir une base client stable et engagée.

C'est dans ce contexte que s'inscrit ce projet, qui vise à exploiter les données issues des usages télécom afin de prédire les risques de résiliation et de fournir des outils d'aide à la décision sous forme de tableaux de bord interactifs. Grâce à la méthodologie CRISP-DM (Cross Industry Standard Process for Data Mining), le projet suit une approche structurée de la data science, allant de la compréhension métier à la modélisation prédictive, en passant par la préparation des données et l'analyse exploratoire.

L'objectif final est double :

- Anticiper les comportements à risque grâce à des modèles de prévision performants ;
- Faciliter l'interprétation et le suivi des indicateurs clés via des visualisations claires et dynamiques.

1. Présentation du projet

Introduction

Dans un environnement télécom particulièrement compétitif, la maîtrise des comportements clients est un enjeu crucial pour réduire le taux de résiliation et optimiser la fidélisation. Ce projet s'appuie sur l'analyse d'un jeu de données riche, contenant des informations contractuelles, financières et comportementales relatives aux clients, incluant notamment un indicateur de résiliation (churn).

1.1 Contexte du projet

Le secteur des télécommunications fait face à des défis majeurs, tels que la fidélisation des clients, la réduction du churn et l'optimisation des services proposés. Avec l'évolution des technologies et la multiplication des offres, la compréhension précise des comportements clients est devenue essentielle pour maintenir un avantage concurrentiel.

Les données clients jouent un rôle croissant dans cette démarche. Le jeu de données utilisé dans ce projet regroupe diverses informations : données contractuelles, financières et comportementales, avec des variables clés telles que l'identifiant client, les volumes de consommation, les revenus générés, ainsi que des indicateurs d'usage entrant et sortant. Chaque enregistrement correspond à un client ou à un contrat unique.

La problématique générale qui se pose est de savoir comment exploiter efficacement ces données pour améliorer la gestion de la relation client et limiter les résiliations.

1.2 Objectifs

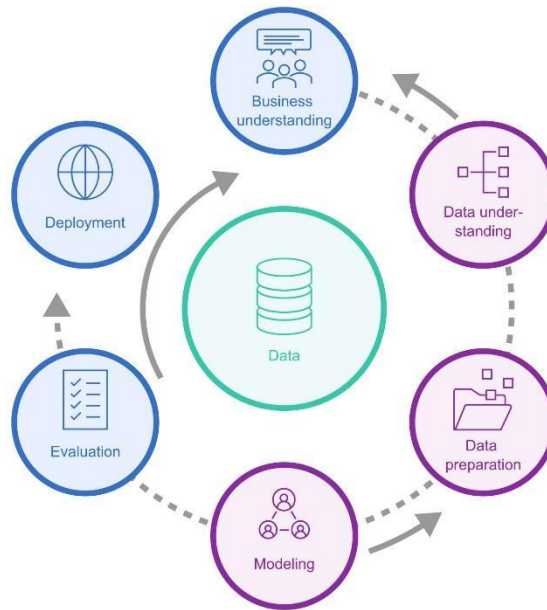
L'objectif principal de ce projet est d'analyser le comportement des clients à partir des données disponibles afin d'en dégager des enseignements pertinents.

Plus précisément, il s'agit de :

- Identifier les facteurs clés liés à la résiliation des contrats (churn).
- Segmenter les clients selon leur profil d'utilisation.
- Développer des outils d'aide à la décision fondés sur l'analyse des données.
- Proposer des recommandations pour améliorer la fidélisation et optimiser les stratégies commerciales.

1.3 Méthodologie adoptée

Pour mener à bien ce projet, nous avons suivi la méthodologie **CRISP-DM** (Cross-Industry Standard Process for Data Mining), largement utilisée dans les projets d'analyse de données. Elle offre une approche structurée en six phases, permettant de transformer des données brutes en connaissances exploitables :



- Compréhension du métier (Business Understanding) : Cette première étape consiste à définir les objectifs métier, à comprendre le contexte de la problématique du churn et à formuler des questions auxquelles l'analyse devra répondre.
- Compréhension des données (Data Understanding) : Elle inclut l'exploration initiale du jeu de données, l'identification des variables clés et la détection de problèmes potentiels (valeurs manquantes, incohérences, etc.).
- Préparation des données (Data Preparation) : Cette phase comprend le nettoyage des données, le traitement des valeurs manquantes, l'encodage des variables catégorielles et la normalisation des variables quantitatives. Elle aboutit à un jeu de données prêt pour l'analyse.
- Modélisation (Modeling) : Des modèles statistiques et d'apprentissage automatique sont construits afin de détecter les profils à risque de résiliation et de segmenter les clients selon leurs comportements.
- Évaluation (Evaluation) : Les modèles sont évalués à l'aide de métriques appropriées (ex. : accuracy, recall, RMSE), afin de s'assurer de leur pertinence par rapport aux objectifs définis au départ.
- Déploiement (Deployment) : Enfin, les résultats sont restitués sous forme de tableaux de bord et de recommandations, afin d'être exploités par les décideurs pour améliorer la stratégie de fidélisation.

Cette méthodologie nous a permis de structurer l'approche analytique tout en gardant une orientation métier claire, essentielle dans le secteur des télécommunications où les décisions doivent être rapidement actionnables.

2. Nettoyage et préparation des données

Introduction

Avant d'analyser les données, il faut s'assurer qu'elles sont propres et correctes. Les données brutes peuvent contenir des erreurs, des informations manquantes ou inutiles. Le nettoyage et la préparation des données permettent de corriger ces problèmes, ce qui aide à obtenir des résultats fiables.

Dans cette partie, nous expliquons les étapes pour transformer les données brutes en un jeu de données prêt à être utilisé : nettoyage, gestion des valeurs manquantes, transformation des catégories, normalisation des chiffres, et sauvegarde finale.

2.1 Nettoyage des données

Le nettoyage initial du jeu de données est une étape fondamentale pour garantir l'intégrité des analyses à venir. Il a été réalisé selon les axes suivants :

a. Suppression des doublons

Une vérification a permis d'identifier et de supprimer les lignes en double, évitant ainsi toute redondance pouvant biaiser les statistiques ou l'entraînement de modèles prédictifs.

b. Élimination des colonnes non pertinentes

Certaines variables ont été retirées du jeu de données, notamment celles :

- Dont les valeurs étaient constantes ou quasi constantes,
- Qui n'étaient pas nécessaires à l'étude,
- Ou qui étaient redondantes avec d'autres variables plus informatives.

c. Uniformisation des formats

Les formats de données ont été standardisés :

- Les dates ont été converties au format datetime ;
- Les variables numériques mal interprétées comme chaînes ont été converties en float ou int selon le cas ;
- Les identifiants et codes ont été traités comme des chaînes de caractères pour éviter toute conversion numérique erronée.

d. Nettoyage des chaînes de caractères

Les colonnes textuelles ont été harmonisées par :

- La mise en minuscules,
- La suppression des caractères spéciaux et des espaces superflus,
- L'uniformisation des libellés (ex. : “Prepaid”, “prepaid”, “PREPAID” → “prepaid”).

e. Filtrage des valeurs incohérentes

Des règles de validation ont permis de détecter et corriger/supprimer :

- Des âges ou volumes négatifs,
- Des codes ou types d'entité inexistant dans les référentiels attendus,
- Des valeurs extrêmes ne respectant pas les bornes métier.

f. Réindexation

Après ces opérations, l'index du DataFrame a été réinitialisé pour assurer une structure claire et continue.

2.2 Traitement des valeurs manquantes

a. Colonnes numériques

Plusieurs colonnes à nature quantitative présentaient des valeurs manquantes (NaN), notamment :

total_nb_recharge, total_rechage, total_u_data, total_rev_option, total_rev_sos, total_u_out, total_u_in, usage_op1, usage_op2, usage_op3, nb_cont_out, nb_cont_in, nb_cell_visite_out, nb_cell_visite_in, nbr_contrat.

Ces variables correspondent à des indicateurs d'usage, de consommation ou de volume.

Les valeurs manquantes ont été remplacées par la valeur 0, indiquant l'absence d'activité ou de transaction sur la période.

Justification :

- La valeur 0 est interprétable (pas d'utilisation),
- Elle évite les erreurs lors de calculs ou d'agrégation,
- Elle permet une meilleure compatibilité avec les modèles statistiques ou machine learning.

b. Colonnes catégorielles

Deux colonnes catégorielles présentaient des valeurs manquantes :
entity_code et entity_type_name.

Les valeurs manquantes ont été remplacées par la modalité la plus fréquente (le mode) :

- entity_code → {mode_entity_code}
- entity_type_name → {mode_entity_type}

Justification :

- Le mode évite l'introduction de modalités arbitraires comme "UNKNOWN",
- Il respecte la distribution réelle des données,
- Il limite la dispersion inutile lors de l'encodage.

❖ *(Remplacer {mode_entity_code} et {mode_entity_type} par les vraies modalités une fois connues.)*

c. Vérification post-traitement

Une vérification finale a confirmé qu'aucune valeur manquante ne subsistait dans les colonnes traitées. Des visualisations avant/après ont été générées pour documenter les changements.

2.3 Encodage des variables catégorielles

Pour permettre l'utilisation des colonnes catégorielles dans des algorithmes numériques (comme les modèles prédictifs), un encodage a été réalisé :

- Les colonnes avec un nombre limité de modalités ont été encodées par Label Encoding, transformant chaque catégorie en un entier unique.
- Pour certaines colonnes plus complexes ou utilisées dans des modèles sensibles aux relations d'ordre, un One-Hot Encoding a été appliqué pour générer une colonne binaire par modalité.

Cet encodage permet :

- Une compatibilité totale avec les algorithmes de machine learning,
- Une meilleure expressivité des variables catégorielles,
- Une réduction des biais introduits par des classements arbitraires.

2.4 Mise à l'échelle des variables (si nécessaire)

Dans les cas où des algorithmes sensibles à l'échelle des données ont été utilisés (comme la régression logistique, les k-means, etc.), une normalisation ou standardisation a été appliquée aux variables numériques :

- Standardisation (*z-score*) : centrage-réduction autour de la moyenne.
- Min-Max Scaling : mise à l'échelle entre 0 et 1.

2.5 Sauvegarde du dataset traité

Le dataset final, nettoyé et prêt à être utilisé pour l'analyse ou la modélisation, a été exporté au format Excel (.xlsx) sous le nom `df1_traité.xlsx`, garantissant sa portabilité pour des outils BI ou d'analyse statistique.

Conclusion

L'ensemble de ces étapes a permis d'obtenir un jeu de données propre, complet et exploitable, prêt à être utilisé dans des analyses exploratoires, de la visualisation, ou des modèles prédictifs. Ce nettoyage méthodique est indispensable pour garantir la qualité des résultats et la fiabilité des conclusions à tirer.

3. Préparation et intégration des données

Introduction

Avant de pouvoir réaliser des analyses pertinentes, il est essentiel de disposer de données propres, cohérentes et bien structurées. Cette phase de préparation et d'intégration joue un rôle central dans tout projet d'analyse de données. Elle vise à transformer des données brutes, souvent incomplètes et dispersées, en un jeu de données consolidé et exploitable.

Dans cette partie, nous détaillons les différentes étapes que nous avons suivies pour nettoyer, transformer et intégrer les données clients dans une base PostgreSQL, en nous appuyant notamment sur l'outil Talend.

3.1 Extraction et nettoyage des données

Les données initiales brutes, issues de différentes sources Excel, présentaient plusieurs anomalies courantes telles que :

- Des valeurs manquantes,
- Des doublons,
- Des incohérences de format (par exemple, des dates mal saisies ou des types de données incorrects).

Pour y remédier, les étapes suivantes ont été réalisées :

- Suppression ou imputation des valeurs manquantes, selon la nature du champ et sa criticité.
- Élimination des doublons afin d'éviter les redondances et les erreurs dans les analyses.
- Correction des formats pour garantir l'homogénéité du jeu de données (formats de date, types de colonnes, uniformisation des noms).

Ces opérations ont permis d'obtenir une base de travail plus fiable et mieux structurée.

3.2 Transformation des données avec Talend

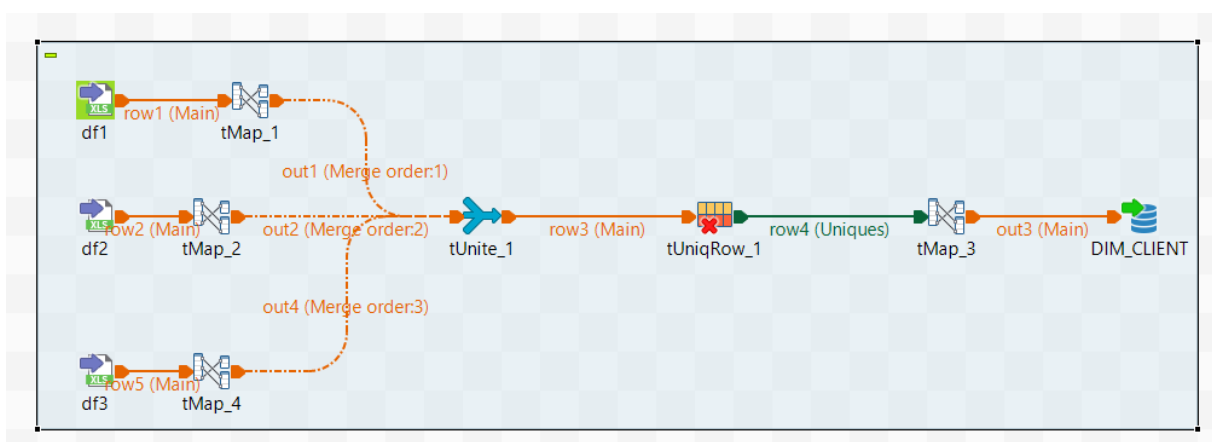
Pour automatiser et fiabiliser le processus de préparation, nous avons utilisé l'outil Talend Open Studio, spécialisé dans les traitements ETL (Extract, Transform, Load). Grâce à sa plateforme graphique et modulaire, il nous a été possible de créer des workflows clairs et reproductibles pour la transformation des données.

3.2.1 Exemple de job Talend : intégration des données clients

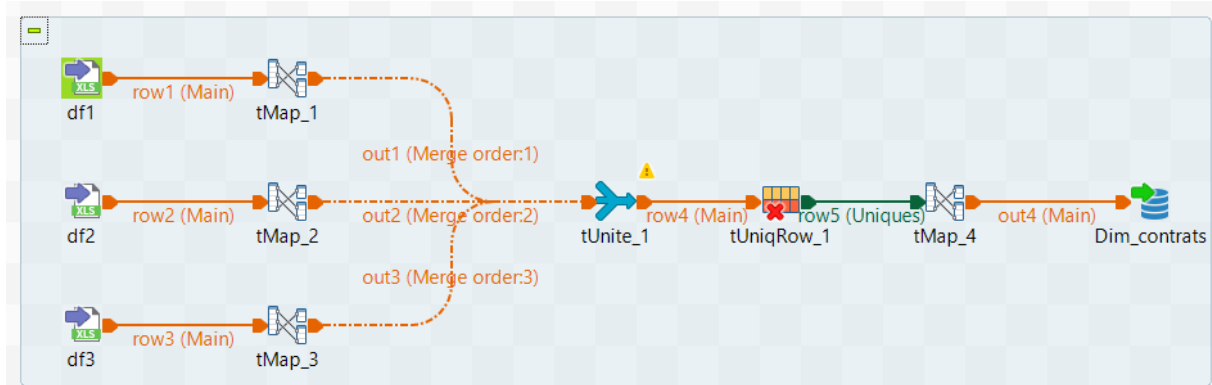
Un job spécifique a été conçu pour consolider les données clients issues de trois fichiers Excel. Ce processus comprend les étapes suivantes :

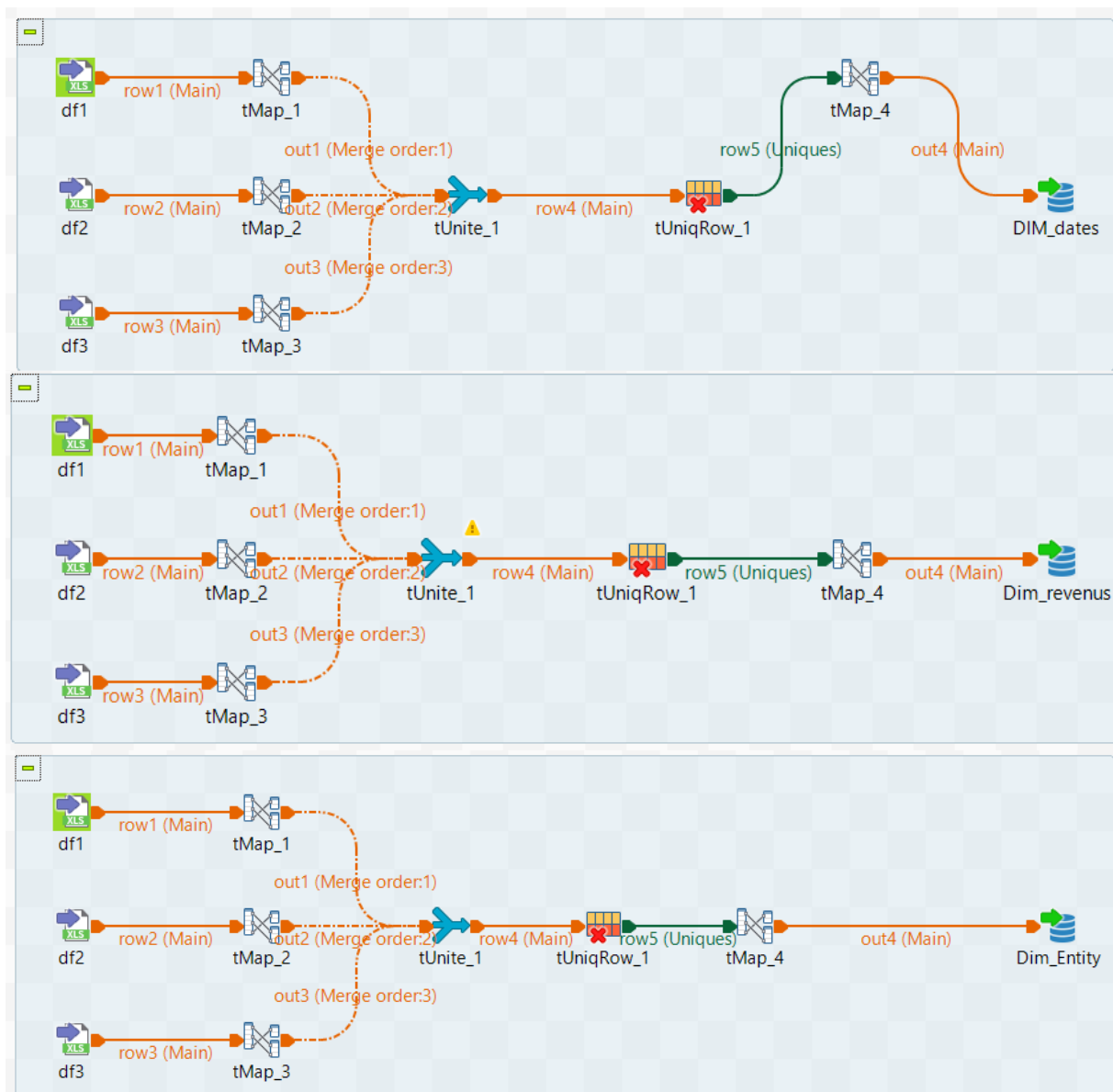
- Lecture des fichiers Excel (df1, df2, df3) via les composants tFileInputExcel, chacun relié à un composant tMap pour effectuer les premières transformations.
- Nettoyage et harmonisation des schémas via tMap_1, tMap_2 et tMap_4 (renommage de colonnes, filtrage des lignes, conversion de types...).
- Fusion des flux avec le composant tUnite_1, qui regroupe les données tout en conservant l'ordre de priorité.
- Suppression des doublons grâce à tUniqRow_1, en se basant sur des clés d'unicité comme l'ID client ou l'adresse email.
- Transformation finale à l'aide de tMap_3, pour adapter les données au modèle cible.
- Chargement dans la base de données PostgreSQL, dans la table DIM_CLIENT, via un composant de sortie.

Ce job permet une intégration fluide, automatisée et traçable des données clients, tout en assurant leur qualité.



De même pour les autres dimensions contrats , dates , revenus et entity.





3.2.2 Construction de la table de faits à l'aide de Talend

Dans le cadre de la modélisation décisionnelle, la table de faits occupe une place centrale, car elle permet la consolidation des indicateurs métiers à analyser. Afin de construire cette table de faits, un job Talend a été mis en place pour automatiser le processus d'intégration des données issues de différentes sources.

Le processus débute par l'extraction de données à partir de trois fichiers Excel contenant des informations opérationnelles. Ces fichiers sont fusionnés à l'aide du composant tUnite, afin de générer un flux unique, qui alimente ensuite une chaîne de transformations successives.

La transformation initiale est réalisée via le composant tMap, qui permet d'harmoniser les structures de données. Par la suite, le flux est enrichi par une série de jointures de

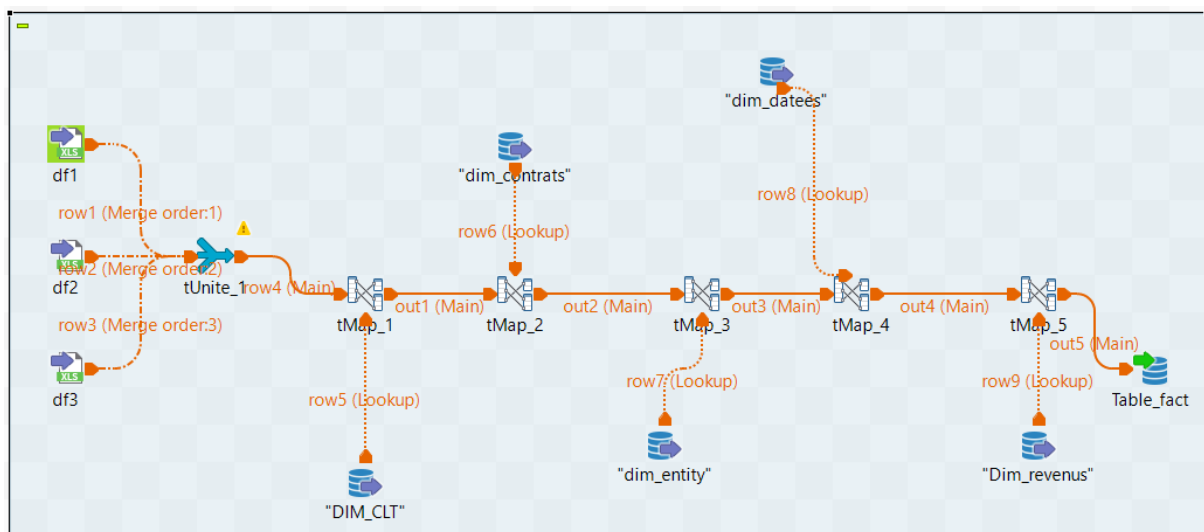
type "lookup" avec plusieurs dimensions de référence préexistantes dans la base de données relationnelle :

- La dimension client (DIM_CLT) pour récupérer les identifiants et attributs clients,
- La dimension contrats (dim_contrats) pour rattacher les données contractuelles,
- La dimension entité (dim_entity) pour associer les structures organisationnelles,
- La dimension temps (dim_dates) pour contextualiser les données dans le temps,
- La dimension revenus (Dim_revenus) pour intégrer les données financières.

Chaque tMap successif assure la jointure avec une dimension spécifique et permet de restructurer les données avant le chargement final.

À l'issue du processus, les données consolidées sont chargées dans la table Table_fact, qui constitue la table de faits centrale du modèle en étoile. Cette dernière contient des clés étrangères vers les différentes dimensions ainsi que des mesures quantitatives permettant d'effectuer des analyses multidimensionnelles (ventes, revenus, volumes, etc.).

Ce job assure ainsi une intégration robuste, normalisée et automatisée des données, préparant efficacement le terrain pour les analyses décisionnelles futures via des outils comme Power BI.



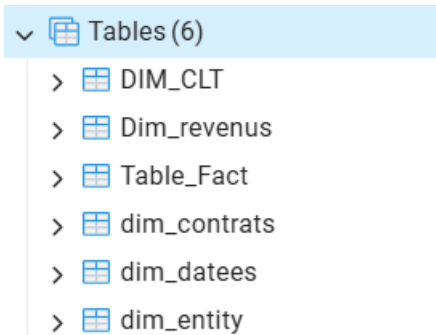
3.3 Chargement dans PostgreSQL

Une fois les données nettoyées et transformées, elles sont chargées dans une base de données PostgreSQL, choisie pour sa fiabilité et ses capacités de gestion de données volumineuses.

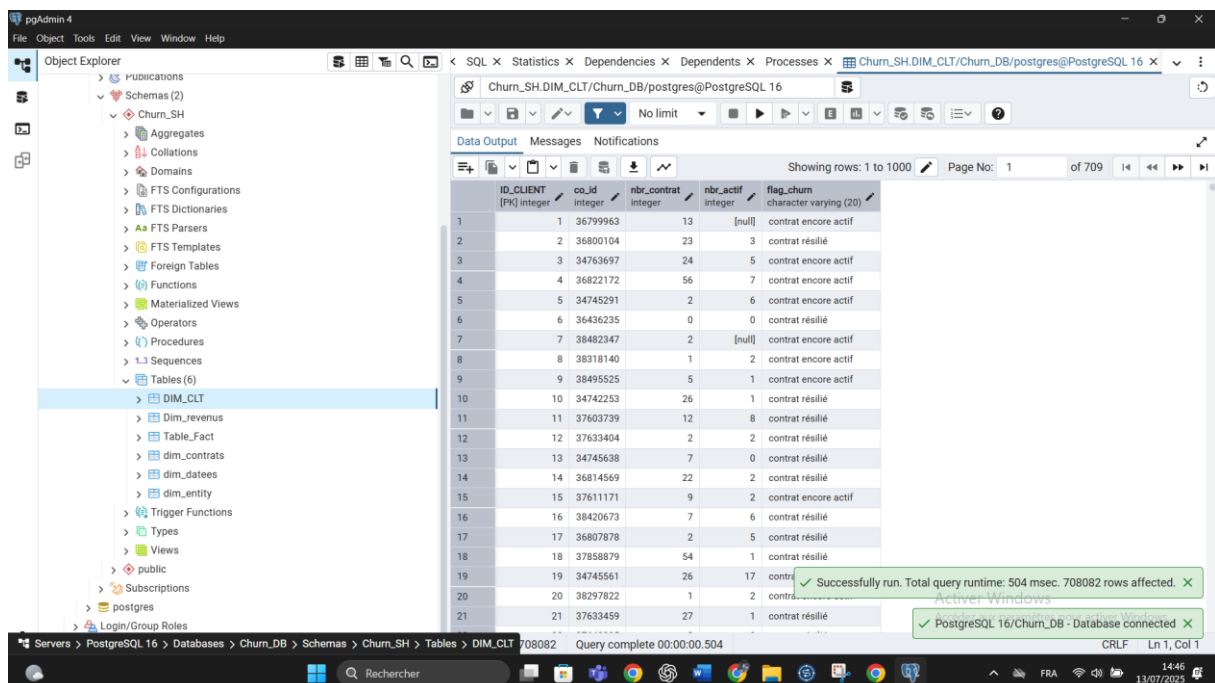
Ce chargement a permis de :

- Structurer les données dans un environnement sécurisé et performant.

- Mettre en place des vues, index et relations facilitant les futures analyses.
- Centraliser les données dans un référentiel unique, exploitable par les outils de Business Intelligence comme Power BI.



Les tableaux sont créés dans PgAdmin4.



Conclusion

Cette étape de préparation et d'intégration a permis de poser des bases solides pour les analyses ultérieures. Grâce à une combinaison d'outils performants et de bonnes pratiques en matière de traitement de données, nous avons obtenu un dataset propre, unifié et fiable.

Le recours à Talend a permis d'automatiser l'ensemble du processus ETL, tout en garantissant la traçabilité et la reproductibilité des flux.

Enfin, le chargement des données dans PostgreSQL a assuré leur accessibilité, sécurité et performance, en vue des explorations analytiques à venir.

4. Création d'un tableau de bord avec Power BI

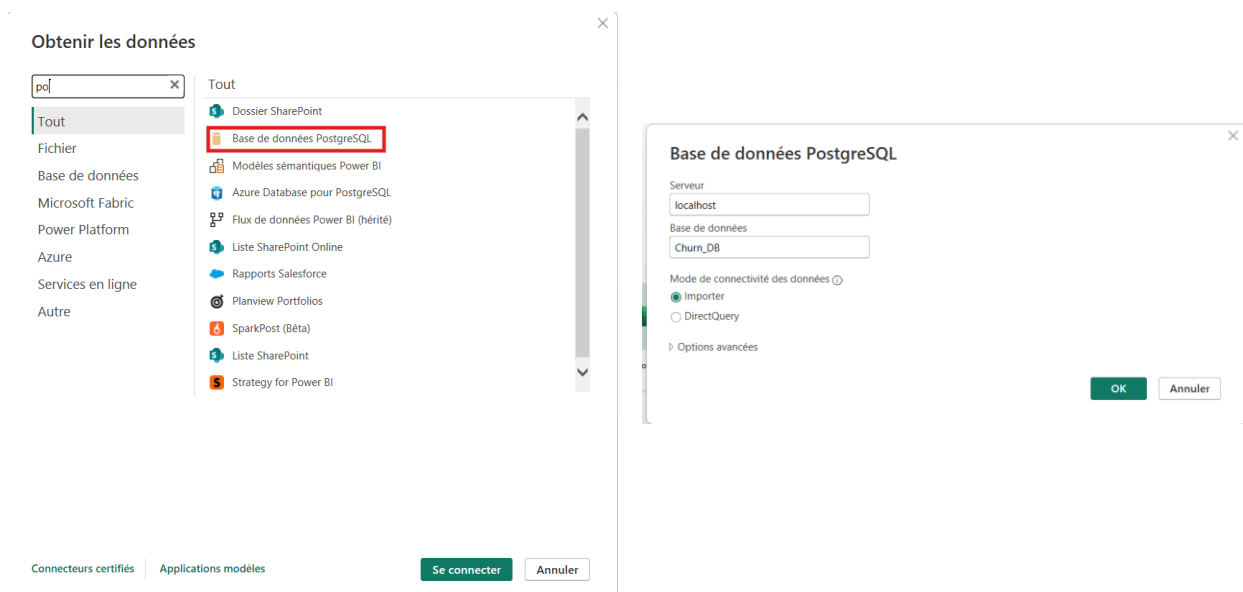
Introduction

Dans le cadre de ce projet, la création d'un tableau de bord interactif et dynamique constitue une étape essentielle pour faciliter l'analyse et la prise de décision. Power BI, outil de Business Intelligence puissant et intuitif, a été choisi pour visualiser les données extraites et préparées dans les étapes précédentes. Ce chapitre présente la démarche adoptée pour connecter Power BI à la base de données PostgreSQL, concevoir un tableau de bord pertinent, analyser les résultats obtenus, et enfin assurer l'automatisation et la mise à jour régulière des données.

4.1 Connexion à la base de données PostgreSQL

La première étape de la création du tableau de bord consiste à établir une connexion sécurisée et efficace entre Power BI et la base de données PostgreSQL qui contient les données nettoyées et intégrées. Power BI propose un connecteur natif permettant de se connecter directement à PostgreSQL via un driver ODBC. Après avoir renseigné les paramètres de connexion tels que l'adresse du serveur, le nom de la base, le port, ainsi que les identifiants d'accès, il est possible d'importer les tables ou d'exécuter des requêtes SQL personnalisées pour récupérer uniquement les données nécessaires à l'analyse.

Cette connexion directe assure l'accès en temps réel aux données actualisées et évite les erreurs liées à l'export manuel.



Navigateur

Options d'affichage ▾

- localhost: Churn_DB [6]
- ☒ Churn_SH.DIM_CLT
 - ☒ Churn_SH.dim_contrats
 - ☒ Churn_SH.dim_dates
 - ☒ Churn_SH.dim_entity
 - ☒ Churn_SH.Dim_revenus
 - ☒ Churn_SH.Table_Fact

Churn_SH.Table_Fact

Aperçu téléchargé le mardi 1 juillet 2025

ID_CLIENT	ID_contrat	ID_entity	ID_date	ID_revenus	total_u_out	total_u_in
1	657874	635973	266697	693318	14,95	0,633
4	706133	648809	266736	708571	13,083	4,183
5	680547	676963	266734	706814	1,967	5,784
7	708843	525194	266723	708590	0	3,333
8	151582	532400	266729	708876	59,766	36,2
9	552040	518334	266727	708904	17,933	8,934
10	687814	691969	266734	693523	5,184	0
11	636062	672628	266730	674080	0,25	5,15
12	706641	676963	266726	706814	0	0
13	704406	691969	266728	706814	3,717	0,1
14	669846	648809	266723	708571	0,366	0
15	225658	575464	266736	708907	0	0
16	553287	531585	266719	16	37,167	15,567
17	226964	634832	266697	708571	13,833	10,95
19	704309	648809	266718	674080	7	0
21	708840	676963	266726	706814	0	0
22	678515	684574	266723	553724	6	3,283
24	698123	635973	266697	708571	10,2	6,3
25	622628	634832	266723	704761	24,216	9,466
27	587496	648809	266726	674080	4,9	12,317
28	701354	691969	266728	706814	0,917	13,917
30	677720	672628	266708	687945	2,583	0,25

Sélectionner les tables associées

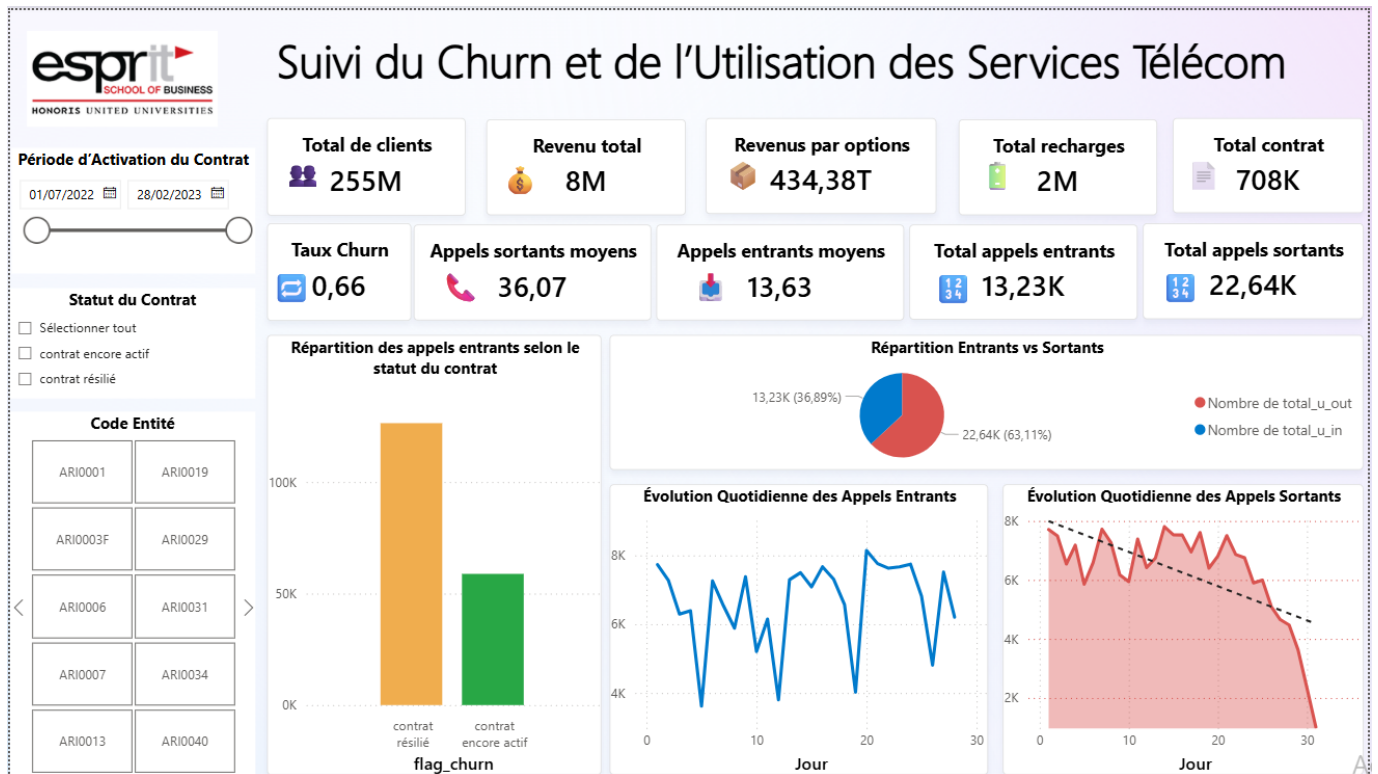
Charger

Transformer les données

Annuler

Act

4.2 Tableau de bord



4.2.1 Mesures Clés Utilisées dans le Tableau de Bord

Dans le cadre de l'analyse du comportement client, plusieurs indicateurs clés de performance ont été définis pour suivre l'évolution de la base client et mesurer la fidélité. Les mesures suivantes ont été intégrées dans Power BI :

- **Nombre total de clients / contrats actifs**

Cet indicateur permet de suivre la taille actuelle de la base client.

Formule : `SOMME (nbr_actif)`

- **Taux de churn (résiliation)**

Ce taux exprime la proportion de clients ayant résilié leur contrat sur l'ensemble des clients. Il s'agit d'un indicateur central dans l'analyse de fidélité.

Formule :

Taux de churn = Nombre total de clients / Nombre de clients résiliés

- **Montant total rechargé (revenu total)**

Mesure les revenus générés par les clients via leurs recharges.

Formule : `SOMME (total_recharge)`

- **Revenus générés par options**

Évalue la contribution des services supplémentaires (options payantes).

Formule : `SOMME (total_rev_option)`

- **Nombre total de recharges**

Permet de suivre la fréquence globale de rechargement.

Formule : `SOMME (total_nb_recharge)`

4.2.2 Méthodologie de Calcul du Taux de Churn à partir d'une Colonne Texte

Les données brutes fournies contiennent une variable textuelle `flag_churn` indiquant le statut du contrat via les libellés "**contrat encore actif**" et "**contrat résilié**". Pour exploiter cette information dans des calculs DAX, une transformation a été nécessaire.

- **Étape 1 : Création d'un indicateur numérique**

```
ChurnFlag =  
IF(  
    'NomDeTaTable'[flag_churn] = "contrat résilié",  
    1,  
    0  
)
```

- **Étape 2 : Définition des mesures intermédiaires**

```
Nb_Total_Contrats = COUNTROWS('NomDeTaTable')
```

```
Nb_Contrats_Resilies =  
CALCULATE(  
    COUNTROWS('NomDeTaTable'),  
    'NomDeTaTable'[ChurnFlag] = 1  
)
```

- **Étape 3 : Calcul du taux de churn**

```
Taux_Churn =  
  
DIVIDE(  
  
    [Nb_Contrats_Resilies],  
  
    [Nb_Total_Contrats],
```

4.2.3 Visualisations

Afin d'exploiter visuellement les données clients et faciliter l'interprétation des indicateurs, plusieurs visualisations interactives ont été intégrées au tableau de bord. Chaque graphique a été conçu pour répondre à un objectif analytique précis, avec un choix de couleurs cohérent facilitant la lecture.

➤ Histogramme – Répartition des appels entrants selon le statut du contrat

Ce graphique en barres met en évidence la distribution des appels entrants en fonction du **statut du contrat** (contrat encore actif vs contrat résilié).

- Les clients ayant résilié sont représentés en **orange**, symbolisant l'arrêt ou la perte de lien avec l'entreprise.
- Les contrats encore actifs sont affichés en **vert**, traduisant la stabilité et la fidélité.

Cette visualisation permet d'observer si les clients résiliés avaient un comportement d'appel différent, ce qui peut aider à anticiper les départs.

➤ Diagramme circulaire – Répartition des appels entrants vs sortants

Le **camembert** met en évidence la part relative des appels **entrants** et **sortants** dans le volume global de communication.

- Les **appels sortants** sont représentés en **rouge**, ce qui attire visuellement l'attention sur l'effort fourni par l'entreprise pour contacter ses clients.
- Les **appels entrants** sont en **bleu**, représentant l'initiative du client.

Ce graphique permet d'évaluer l'équilibre entre interaction client proactive et réactive, un aspect essentiel dans la qualité de la relation client.

➤ Courbe – Évolution quotidienne des appels entrants

Cette courbe suit le nombre d'**appels entrants** jour par jour sur une période donnée. Elle permet de détecter des **pics d'activité**, des **baisses anormales**, ou des **tendances saisonnières**. Une ligne fluide bleue permet de visualiser les variations de l'intérêt client ou de la dépendance au service.

➤ Graphique en aires – Évolution quotidienne des appels sortants

Ce graphique présente l'évolution des **appels sortants** avec une **aire colorée en rouge** et une **tendance régressive** visible à travers une ligne de tendance pointillée.

Cette représentation facilite la détection d'une **baisse progressive de l'activité sortante**, qui peut signaler :

- une perte d'engagement des équipes internes,
- un changement de stratégie commerciale,
- ou une baisse globale de la base client.

4.2.4 Filtres Interactifs

Trois filtres dynamiques ont été intégrés pour permettre une analyse ciblée et personnalisée des données :

- **Période d'Activation du Contrat**

Permet de sélectionner une plage de dates afin de n'analyser que les contrats activés durant une période donnée. Ce filtre est utile pour observer les évolutions dans le temps ou isoler des campagnes spécifiques.

- **Statut du Contrat**

Filtre binaire permettant de distinguer les **contrats encore actifs** des **contrats résiliés**. Il facilite l'analyse comportementale en fonction de la fidélité du client.

- **Code Entité**

Permet de filtrer les données par entité commerciale, agence ou zone géographique. Cela aide à comparer les performances d'une entité à une autre et à détecter les écarts de comportement.

Conclusion

La création du tableau de bord avec Power BI a permis de transformer les données brutes en informations visuelles et exploitables. La connexion directe à la base PostgreSQL, une conception soignée centrée sur les besoins utilisateurs, ainsi que l'automatisation de l'actualisation des données, contribuent à un outil décisionnel performant. Ce tableau de bord facilite non seulement l'analyse en temps réel, mais constitue aussi un levier essentiel pour améliorer la gestion et la prise de décision au sein de l'organisation.

5. Développement d'un Modèle de Prédiction du Churn Client

Introduction

Dans un contexte où la rétention client constitue un enjeu stratégique majeur pour les entreprises, la capacité à anticiper le comportement de désengagement des clients devient essentielle. L'objectif de cette section est de concevoir et d'évaluer des modèles de classification permettant de prédire si un client est susceptible de résilier son contrat. Le problème est formalisé comme une classification binaire avec la variable cible `flag_churn`, où:

- `1` indique un contrat résilié,
- `0` un contrat encore actif.

Le processus suivi s'articule en plusieurs étapes : préparation et exploration des données, traitement des valeurs extrêmes, analyse bivariable, préparation du jeu de données pour l'entraînement, choix et comparaison de plusieurs modèles de classification.

5.1 Analyse exploratoire et préparation des données

5.1.1 Description du jeu de données

Le jeu de données contient 219^[1]26 observations et 18 variables. Il couvre un ensemble de comportements clientèles (nombre de recharges, volume de données consommées, nombre de contacts sortants, etc.) et des caractéristiques de rattachement organisationnel (code entité, type d'agence, etc.).

Les principales variables explicatives quantitatives incluent :

- `total_u_data` : volume total de données consommées,
- `total_nb_recharge`, `total_recharge` : fréquence et volume de recharges,
- `nb_cont_out`, `nb_cont_in` : activité en appel sortant/entrant,
- `nbr_contrat`, `nbr_actif` : nombre de contrats rattachés à l'entité.

5.1.2 Qualité des données

- Valeurs manquantes : aucune valeur manquante recensée dans l'ensemble du dataset.
- Doublons : aucun doublon identifié.
- Typage des données : cohérent avec les types attendus (int, float, datetime, object).

► Valeurs manquantes

```
dataset.isnull().sum() / len(dataset)
```

```
co_id          0.0
activation_date 0.0
total_nb_recharge 0.0
total_recharge 0.0
total_u_data    0.0
total_rev_option 0.0
total_u_out     0.0
total_u_in      0.0
usage_op3       0.0
nb_cont_out     0.0
nb_cont_in      0.0
nb_cell_visite_out 0.0
nb_cell_visite_in 0.0
entity_code     0.0
entity_type_name 0.0
nbr_contrat     0.0
nbr_actif       0.0
flag_churn      0.0
dtype: float64
```

► Doublons

```
print(dataset.duplicated().sum())
```

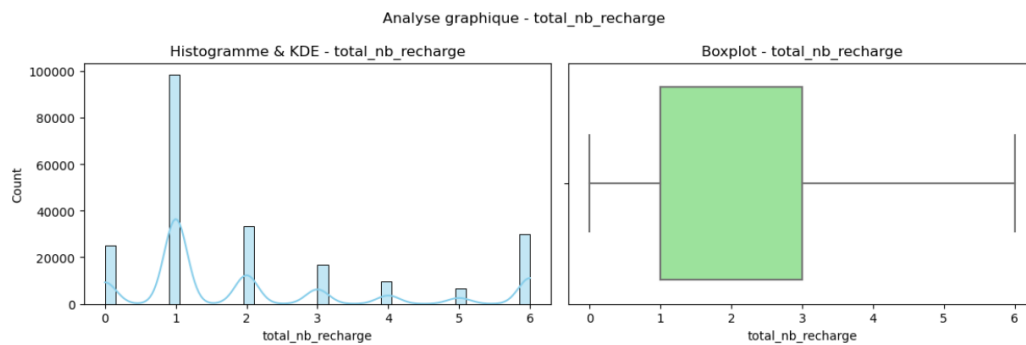
```
0
```

5.2 Analyse statistique univariée

Une exploration statistique a permis d'évaluer la dispersion des variables :

- Les distributions sont majoritairement asymétriques à droite ($\text{skew} > 0$),
- Les tests de normalité (Shapiro-Wilk et D'Agostino) indiquent une non-normalité significative pour toutes les variables quantitatives.
- Les outliers ont été détectés via la méthode IQR, puis traités par winsorisation (plafonnement aux bornes). Cela permet de réduire leur influence sans perte d'information.

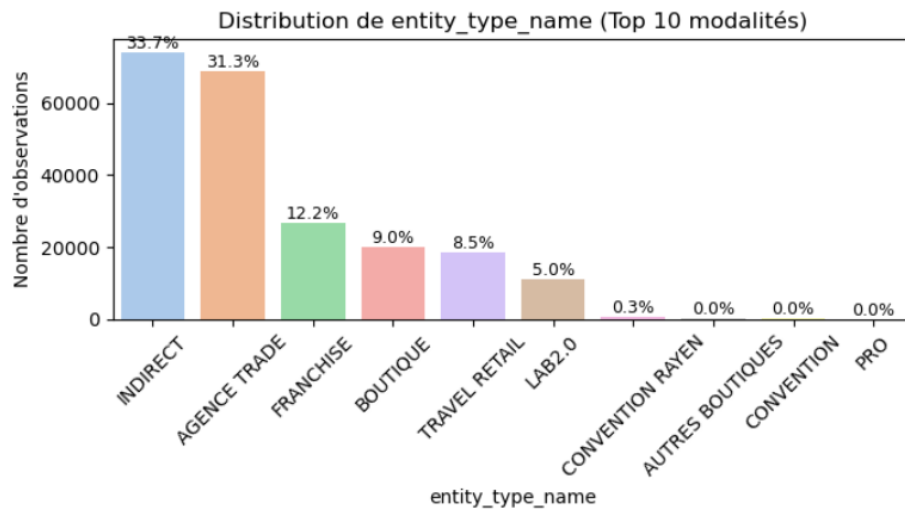
📊 Analyse de la forme - total_nb_recharge
➤ Asymétrie (skewness) : 1.07 → droite
➤ Aplatissement (kurtosis) : -0.18 → platykurtique
➤ Test de Shapiro-Wilk : $p = 0.0000$ (non normale)
➤ Test de D'Agostino : $p = 0.0000$ (non normale)



entity_type_name - Fréquences relatives (%) (top 10) :

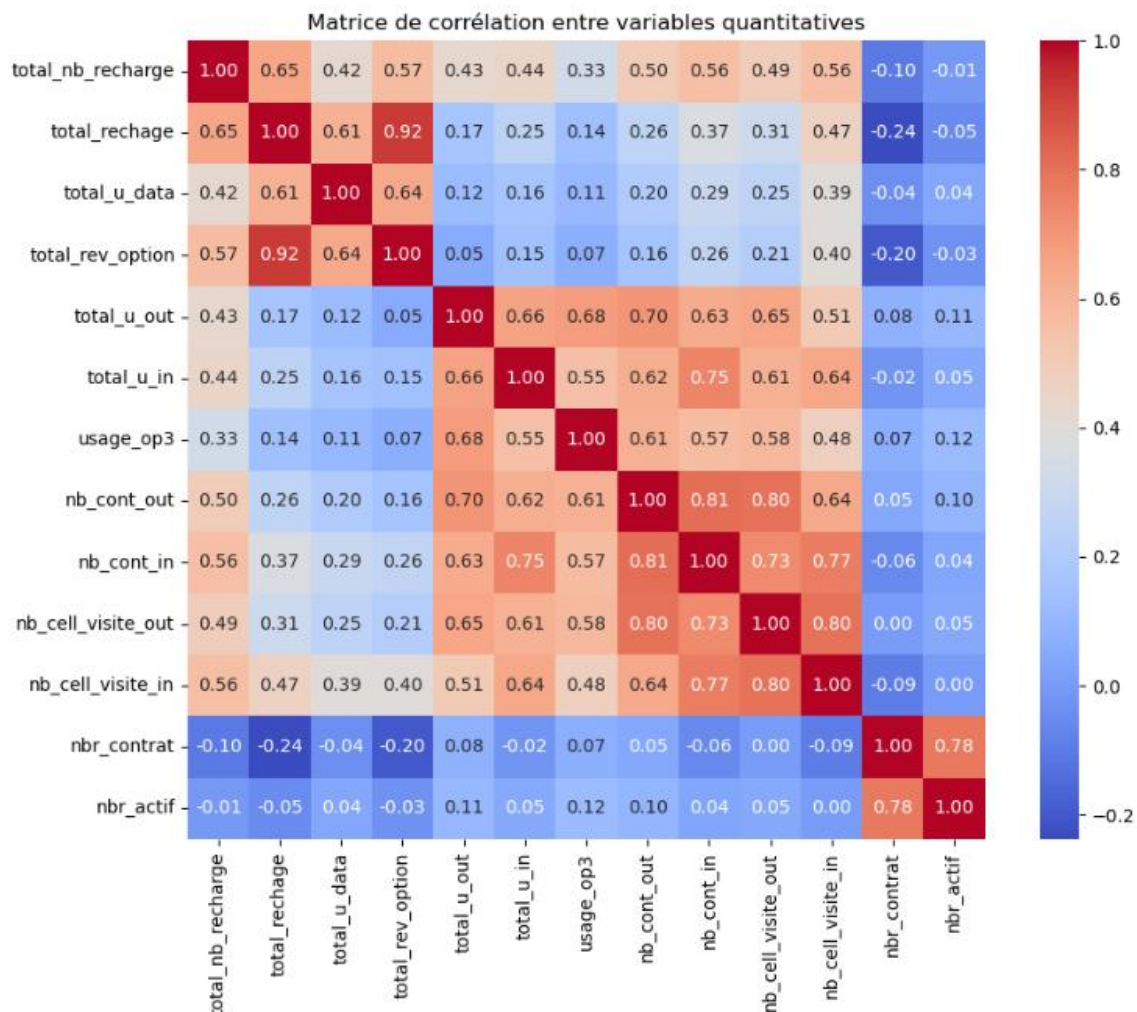
INDIRECT	33.70
AGENCE TRADE	31.35
FRANCHISE	12.20
BOUTIQUE	9.03
TRAVEL RETAIL	8.46
LAB2.0	4.97
CONVENTION RAYEN	0.26
AUTRES BOUTIQUES	0.01
CONVENTION	0.01
PRO	0.01

Name: entity_type_name, dtype: float64

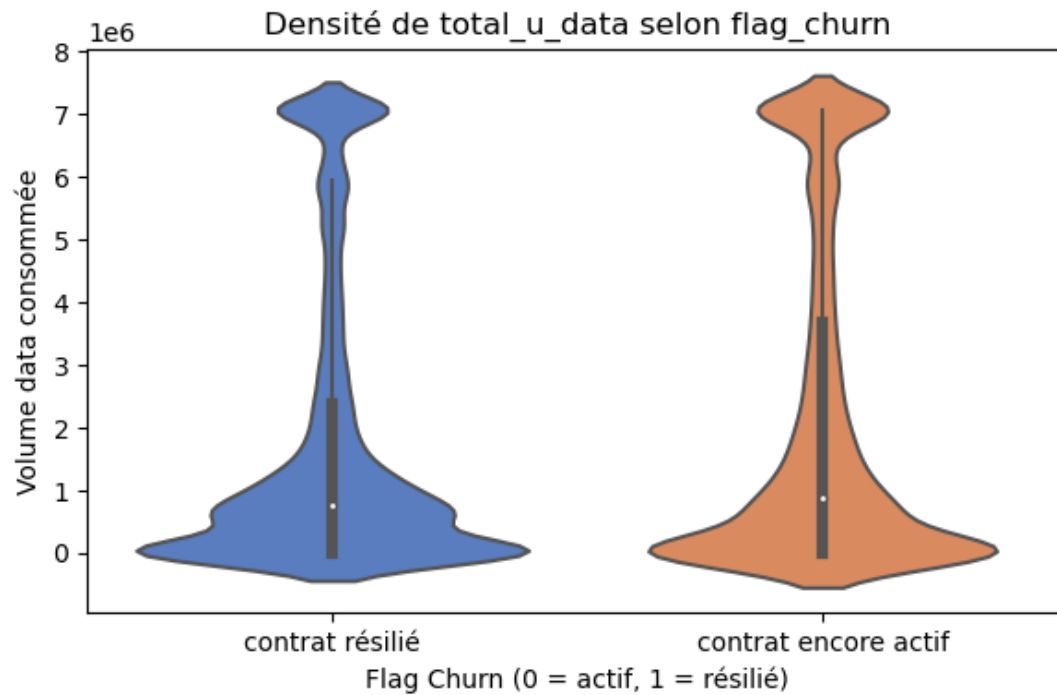


5. 3 Analyse bivariable

Corrélation entre variables quantitatives : matrice de corrélation visualisée via heatmap.



- Quantitatives vs flag_churn : des variables comme total_u_data, nb_cont_out montrent des différences marquées selon le churn.



- Catégorielles vs flag_churn : les résultats du test du khi2 montrent une forte dépendance entre le churn et entity_type_name.

Tableau croisé (%) :

entity_type_name flag_churn	AGENCE TRADE	AUTRES	AUTRES BOUTIQUES	BOUTIQUE	\
contrat encore actif	25.897591	0.001262	0.025249	9.897490	
contrat résilié	34.414070	0.000711	0.008534	8.537557	

entity_type_name flag_churn	CONVENTION	CONVENTION RAYEN	FRANCHISE	INDIRECT	\
contrat encore actif	0.029036	0.611019	16.979751	37.957633	
contrat résilié	0.001422	0.068272	9.501188	31.307693	

entity_type_name flag_churn	LAB2.0	PRO	TRAVEL RETAIL	
contrat encore actif	6.511640	0.012624	2.076706	
contrat résilié	4.104854	0.000711	12.054987	

5.4 Ingénierie des données

5.4.1 Encodage et définition de la variable cible

La variable `flag_churn` a été encodée en binaire (1=résilié, 0=actif). Les 13 variables explicatives retenues sont toutes quantitatives. Un sous-ensemble de données propre est créé pour l'entraînement.

5.4.2 Mise à l'échelle (Scaling)

Une standardisation via `StandardScaler` est appliquée aux variables quantitatives pour assurer une échelle comparable entre features, ce qui est crucial pour les algorithmes basés sur les distances.

```
Avant scaling : [1.00000000e+00 8.78000021e-01 5.63200000e+05 0.00000000e+00
9.40000000e+01 7.21600000e+00 2.11170000e+01 2.30000000e+01
6.00000000e+00 1.30000000e+01 8.00000000e+00 1.00000000e+01
1.00000000e+00]
Après scaling : [-0.5939734 -0.70192943 -0.55492411 -0.67185246 0.80692802 -0.16757674
0.88922367 2.13131028 0.09128151 1.20343196 0.01710967 0.80803785
-0.60665711]
```

5.4.3 Split des données en train/test

Les données sont divisées en :

- 80% pour l'entraînement,
 - 20% pour le test,
- avec stratification pour respecter la proportion des classes.

```
Taille train : (175860, 13)
Taille test  : (43966, 13)
```

```
Répartition dans y_train :
1    0.639662
0    0.360338
dtype: float64
```

```
Répartition dans y_test :
1    0.639653
0    0.360347
dtype: float64
```

5.5 Modélisation et comparaison des algorithmes

5.5.1 Modèle K-Nearest Neighbors (KNN)

Un premier modèle KNN a été entraîné avec $k=5$, puis optimisé par validation croisée (`GridSearchCV`) où le meilleur paramètre est $k=13$.

F1-score KNN optimisé : 0.8130

--- Évaluation modèle KNN optimisé ---

Matrice de confusion :

```
[[ 9353  6490]
```

```
 [ 4417 23706]]
```

Accuracy : 0.7519

Recall : 0.8429

Precision : 0.7851

Rapport de classification :

	precision	recall	f1-score	support
0	0.68	0.59	0.63	15843
1	0.79	0.84	0.81	28123
accuracy			0.75	43966
macro avg	0.73	0.72	0.72	43966
weighted avg	0.75	0.75	0.75	43966

5.5.2 Régression logistique

Malgré une capacité de rappel élevée (0.870), la régression logistique souffre d'une précision inférieure aux modèles non linéaires.

F1-score : 0.7922

--- Évaluation Logistic Regression ---

Matrice de confusion :

```
[[ 6652  9191]
```

```
 [ 3650 24473]]
```

Accuracy : 0.708

Precision : 0.727

Recall : 0.870

Rapport de classification complet :

	precision	recall	f1-score	support
0	0.65	0.42	0.51	15843
1	0.73	0.87	0.79	28123
accuracy			0.71	43966
macro avg	0.69	0.65	0.65	43966
weighted avg	0.70	0.71	0.69	43966

5.5.3 Random Forest

Ce modèle d'ensemble se distingue par sa robustesse et sa capacité à capturer les interactions non linéaires. Il atteint la meilleure performance globale.

F1-score : 0.8284

--- Évaluation Random Forest ---

Matrice de confusion :

```
[[10072  5771]
```

```
 [ 4157 23966]]
```

Accuracy : 0.7742

Recall : 0.8522

Precision : 0.8059

Rapport de classification :

	precision	recall	f1-score	support
0	0.71	0.64	0.67	15843
1	0.81	0.85	0.83	28123
accuracy			0.77	43966
macro avg	0.76	0.74	0.75	43966
weighted avg	0.77	0.77	0.77	43966

5.5.4 Arbre de décision (Decision Tree)

Bien qu'interprétable, l'arbre de décision offre des performances inférieures. Il est cependant utile à des fins explicatives.

F1-score : 0.7611

Comparaison premiers résultats (y_test vs prédiction) :

Vérité : 0 - Prédiction : 0
Vérité : 1 - Prédiction : 1
Vérité : 1 - Prédiction : 1
Vérité : 0 - Prédiction : 1
Vérité : 1 - Prédiction : 1
Vérité : 1 - Prédiction : 1
Vérité : 1 - Prédiction : 1
Vérité : 0 - Prédiction : 1
Vérité : 1 - Prédiction : 1
Vérité : 0 - Prédiction : 0

--- Évaluation Decision Tree ---

Matrice de confusion :

```
[[ 9247  6596]
 [ 6793 21330]]
```

Accuracy : 0.6955

Recall : 0.7585

Precision : 0.7638

Rapport de classification :

	precision	recall	f1-score	support
0	0.58	0.58	0.58	15843
1	0.76	0.76	0.76	28123
accuracy			0.70	43966
macro avg	0.67	0.67	0.67	43966
weighted avg	0.70	0.70	0.70	43966

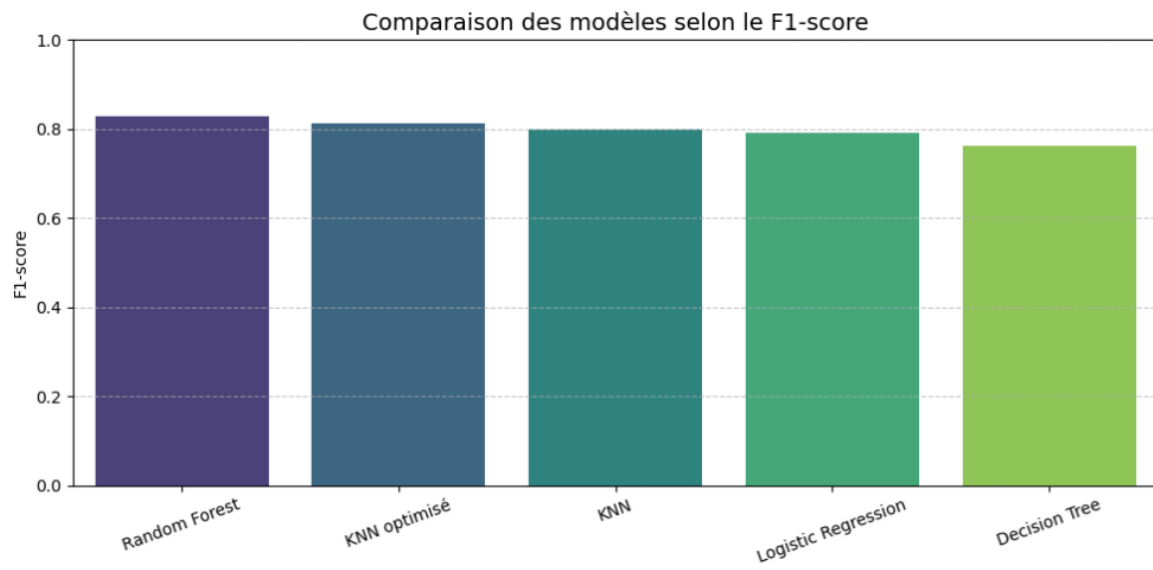
5.6 Synthèse des performances

Un récapitulatif des performances selon les principales métriques est présenté ci-dessous :

Modèle	Accurac y	Precisio n	Recal l	F1- score
Random Forest	0.7742	0.8059	0.8522	0.8284
KNN optimisé	0.7519	0.7851	0.8429	0.8130
KNN simple	0.7365	0.7768	0.8252	0.8003
Régression Logistique	0.7079	0.7270	0.8702	0.7922
Arbre de Décision	0.6955	0.7638	0.7585	0.7611

--- Évaluation finale basée sur le F1-score ---

	Modèle	F1-score
0	Random Forest	0.8284
1	KNN optimisé	0.8130
2	KNN	0.8003
3	Logistic Regression	0.7922
4	Decision Tree	0.7611



Conclusion

Le modèle Random Forest a été retenu comme le plus performant pour la prédiction du churn client. Il combine précision, rappel et robustesse, en s'adaptant à la complexité des données sans prétraitement excessif.

6. Prédiction du churn par analyse de séries temporelles

Introduction

L'anticipation de la résiliation client, appelée communément churn, constitue un enjeu majeur pour les entreprises du secteur des télécommunications. Dans cette partie, nous nous intéressons à la modélisation du churn à travers une approche basée sur l'analyse de séries temporelles. L'objectif est de construire un modèle prédictif robuste capable de détecter les variations temporelles du churn à court terme et de proposer des prévisions fiables.

6.1 Préparation des données

6.1.1 Nettoyage et traitement

Le jeu de données initial comprenait 222 344 lignes et 20 colonnes, avec un taux de valeurs manquantes très variable. Un nettoyage rigoureux a été réalisé selon les étapes suivantes :

- Suppression des variables contenant plus de 90 % de valeurs manquantes (e.g. `total_rev_sos`, `usage_op1`, `usage_op2`).
- Remplacement des valeurs manquantes restantes dans les colonnes quantitatives par des zéros (approche prudente dans un objectif exploratoire).

Transformation de la colonne `activation_date` au format datetime.

6.1.2 Agrégation temporelle

Pour construire une série temporelle continue, la variable `flag_churn` (indiquant si un client a résilié ou non) a été agrégée à une granularité journalière. Le résultat est une série temporelle de la forme :

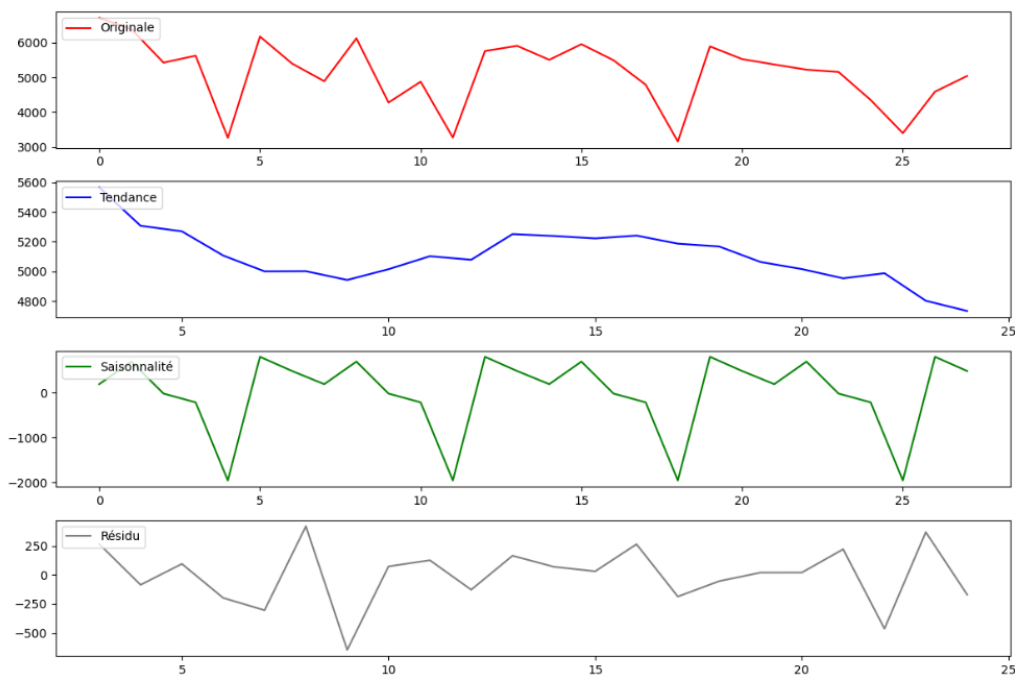
- **Index temporel** : dates comprises entre le 1er février 2023 et le 28 février 2023.
- **Valeur observée** : nombre de clients résiliés par jour.

6.2 Analyse exploratoire de la série temporelle

6.2.1 Composantes de la série

Une décomposition de la série a été réalisée afin d'en extraire les composantes suivantes :

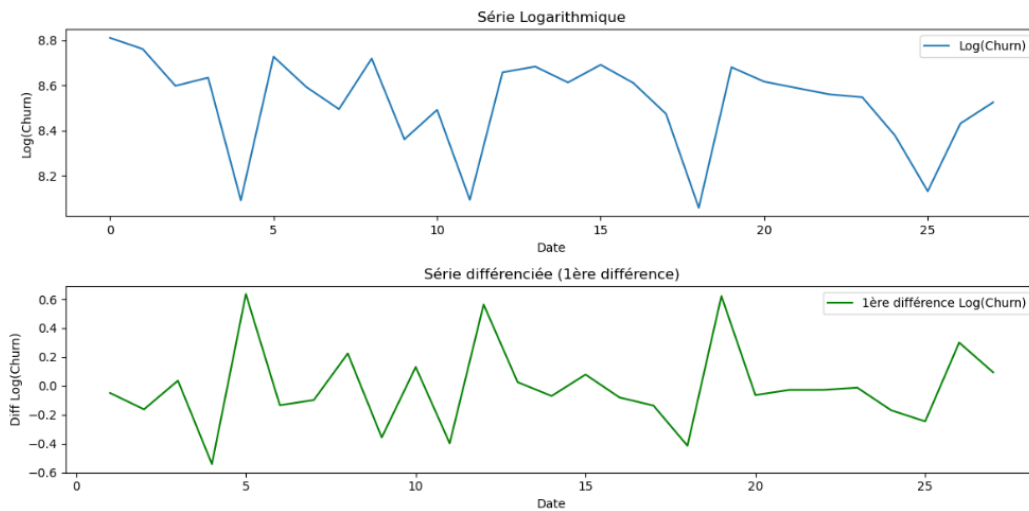
- **Tendance** : faible évolution globale dans le mois observé.
- **Saisonnalité** : récurrence hebdomadaire observée.
- **Résidus** : variations irrégulières et bruits.



6.2.2 Stationnarité de la série

Le test de Dickey-Fuller augmenté (ADF) a été appliqué à différentes versions de la série :

- La série brute s'est révélée **non-stationnaire** (p-value > 0.05).
- Une transformation logarithmique suivie d'une différenciation a permis d'atteindre la stationnarité (p-value < 0.05).



- Une différenciation saisonnière avec une période de 14 jours a également renforcé la stabilité du signal.

ADF Statistic (diff): -8.642845

p-value (diff): 0.000000

Critical Values (diff):

1%: -3.7883858816542486

5%: -3.013097747543462

10%: -2.6463967573696143

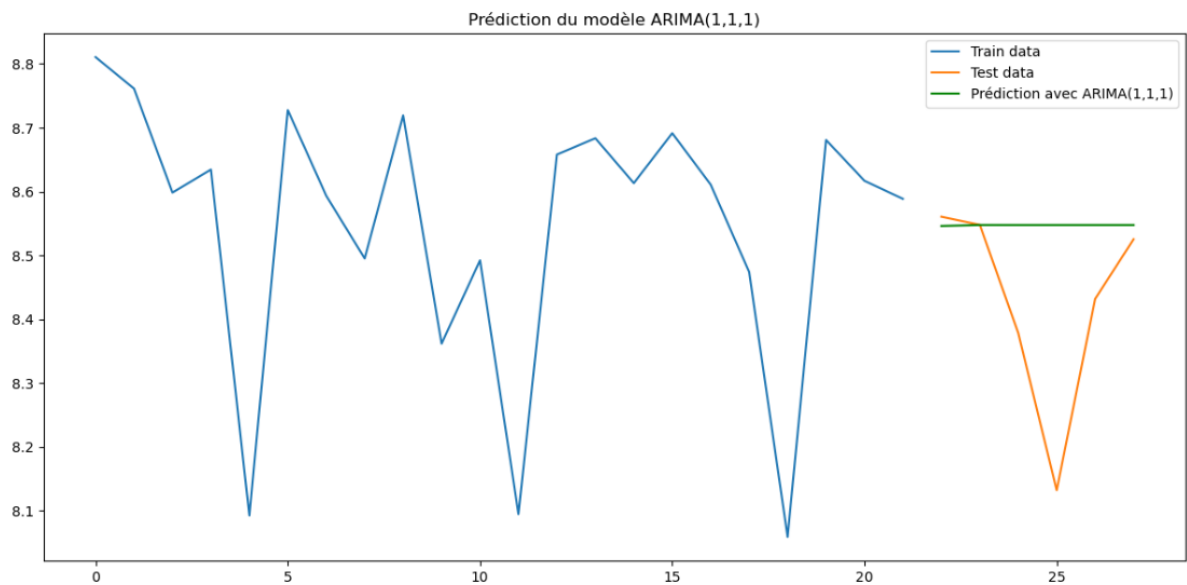
La série différenciée est stationnaire (rejette l'hypothèse nulle).

6.3 Modélisation prédictive

6.3.1 Modèle ARIMA (1,1,1)

Le modèle ARIMA a servi de point de départ pour capter la dynamique temporelle du churn. Il a été entraîné sur 80 % de la série (données de train).

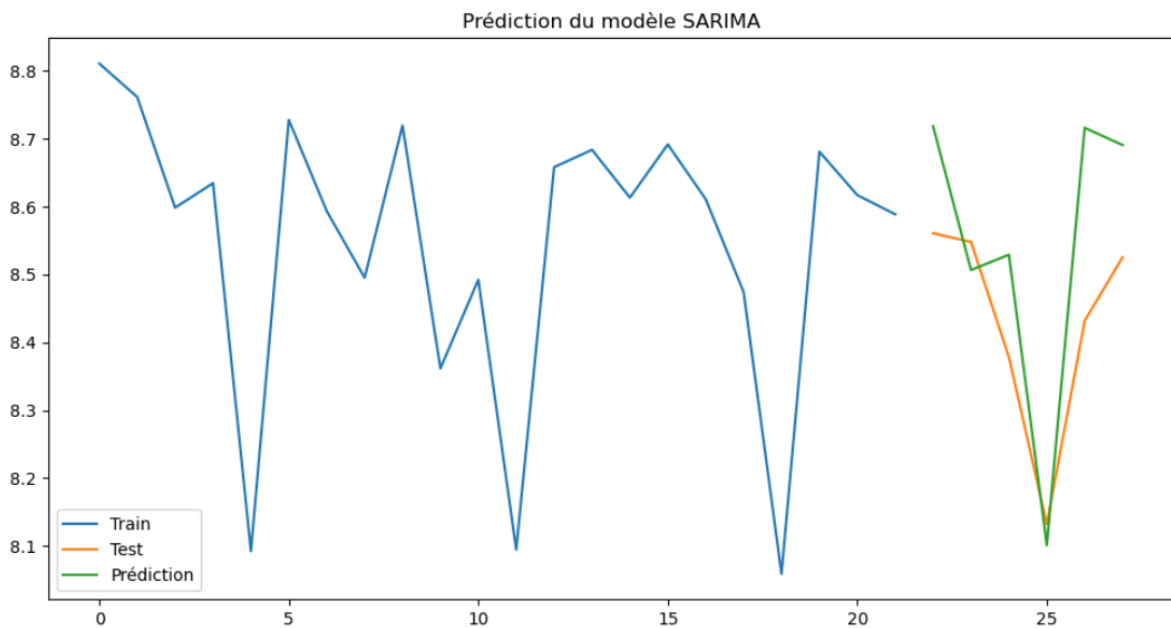
- Le diagnostic des résidus montre une absence significative d'autocorrélation.
- Les performances sur la série de test sont toutefois limitées.



6.3.2 Modèle SARIMA (3,1,0) (1,1,0) [7]

Suite à la détection d'une saisonnalité hebdomadaire, un modèle SARIMA a été testé avec succès. Il a présenté :

- Une meilleure vraisemblance (Log-Likelihood : 14.04).
- Des critères AIC/BIC améliorés par rapport à ARIMA.
- Des résidus blancs validés par le test de Ljung-Box ($p > 0.05$).

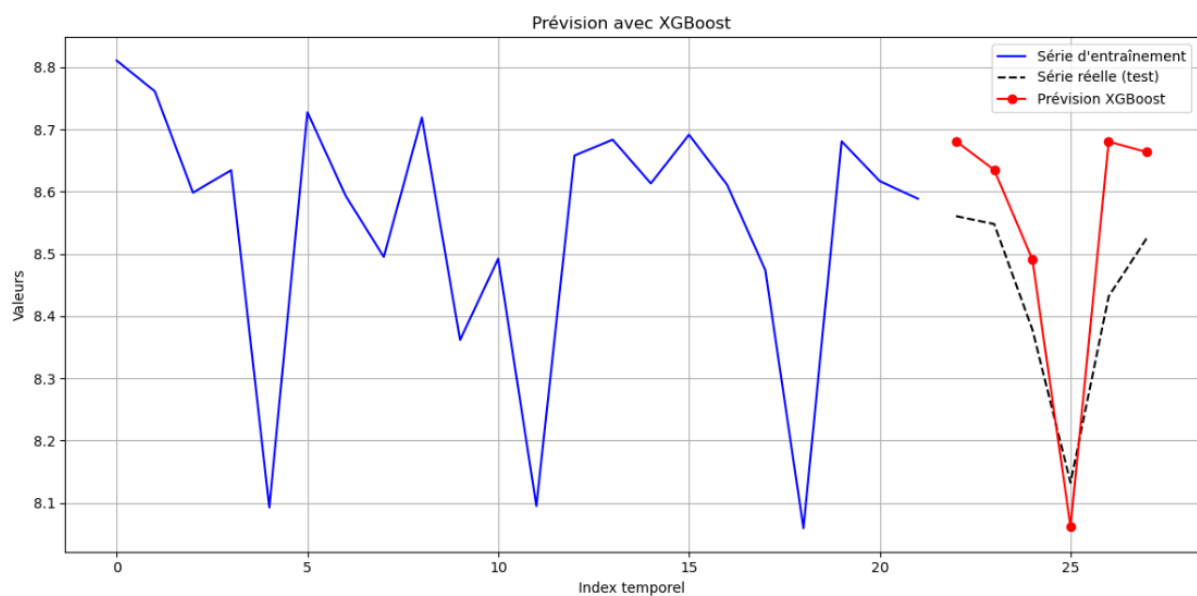


6.3.3 Modèle XGBoost

Le modèle XGBoost a été entraîné sur des fenêtres glissantes (lags) extraites de la série log-transformée.

- Il offre des performances supérieures sur toutes les métriques (MAE, RMSE, MAPE).
- Il est adapté aux comportements non linéaires et irréguliers.

MAE : 0.1295
RMSE : 0.1418
MAPE : 1.53 %



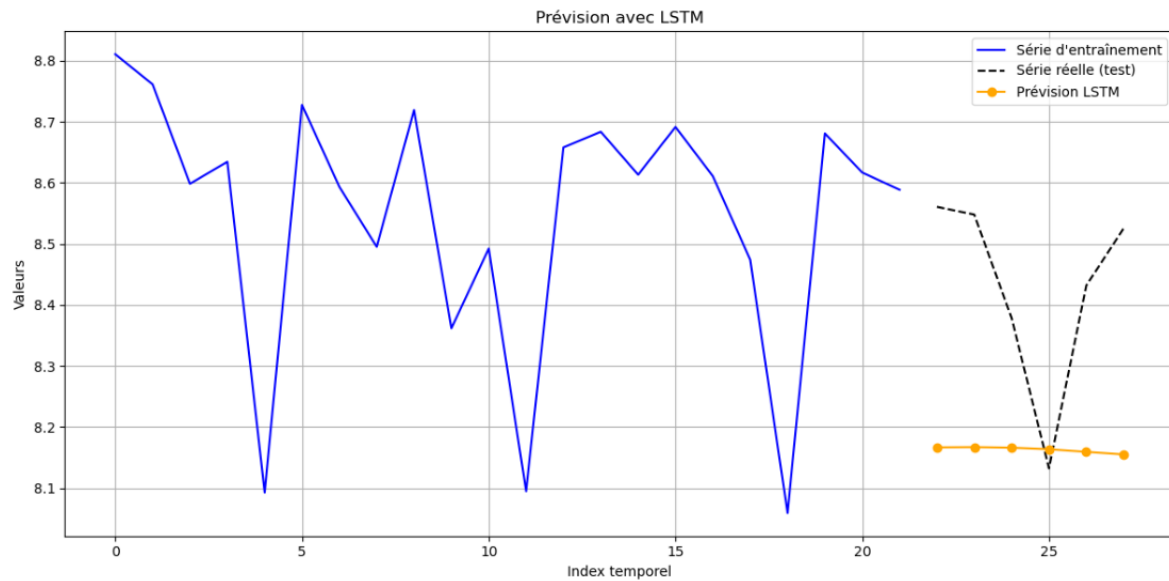
6.3.4 Modèle LSTM

Le réseau de neurones LSTM a été entraîné avec 10 lags sur des données normalisées. Bien que prometteur :

- Le réseau n'a pas surpassé XGBoost sur cette taille de données.

- Il pourrait être amélioré avec un ajustement d'hyperparamètres.

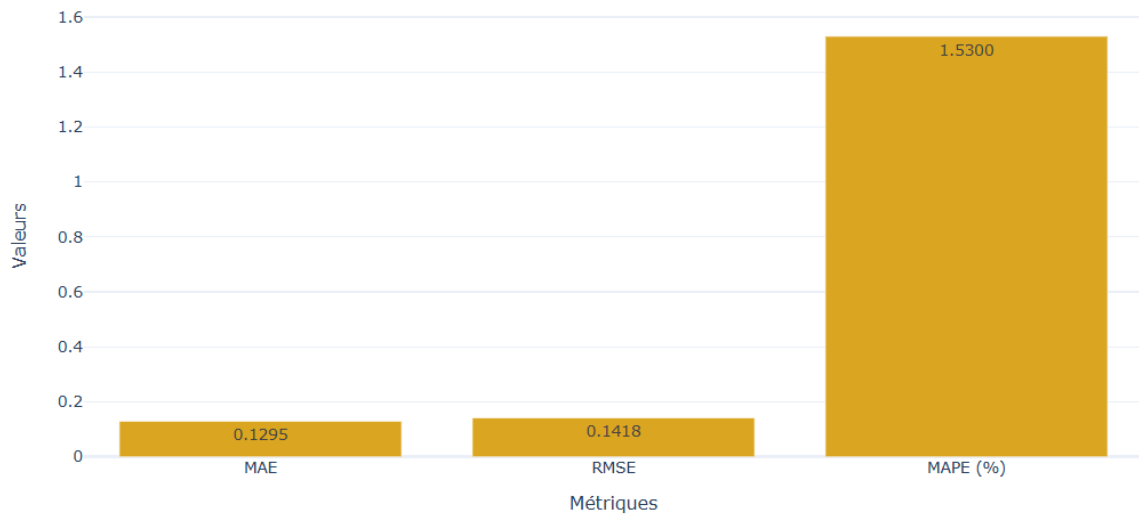
MAE : 0.2769
RMSE : 0.3049
MAPE : 3.26 %



6.4 Évaluation et comparaison des modèles

Modèle	MAE	RMSE	MAPE	Interprétation
SARIMA	0.138 3	0.162 4	1.63	Moins précis, modèle de base intéressant.
XGBoost	0.129 5	0.141 8	1.53	Meilleur modèle global, très performant.
LSTM	0.163 5	0.175 7	1.94	Sensible à la taille de l'échantillon, à affiner.

🎯 Performance du meilleur modèle : XGBoost



Conclusion

L'analyse prédictive du churn à l'aide de séries temporelles a permis d'évaluer l'efficacité de plusieurs modèles : ARIMA, SARIMA, XGBoost et LSTM. Parmi ces derniers, **XGBoost** se distingue par sa précision et sa capacité à capturer des dynamiques complexes, en particulier dans des contextes de séries courtes et bruitées.

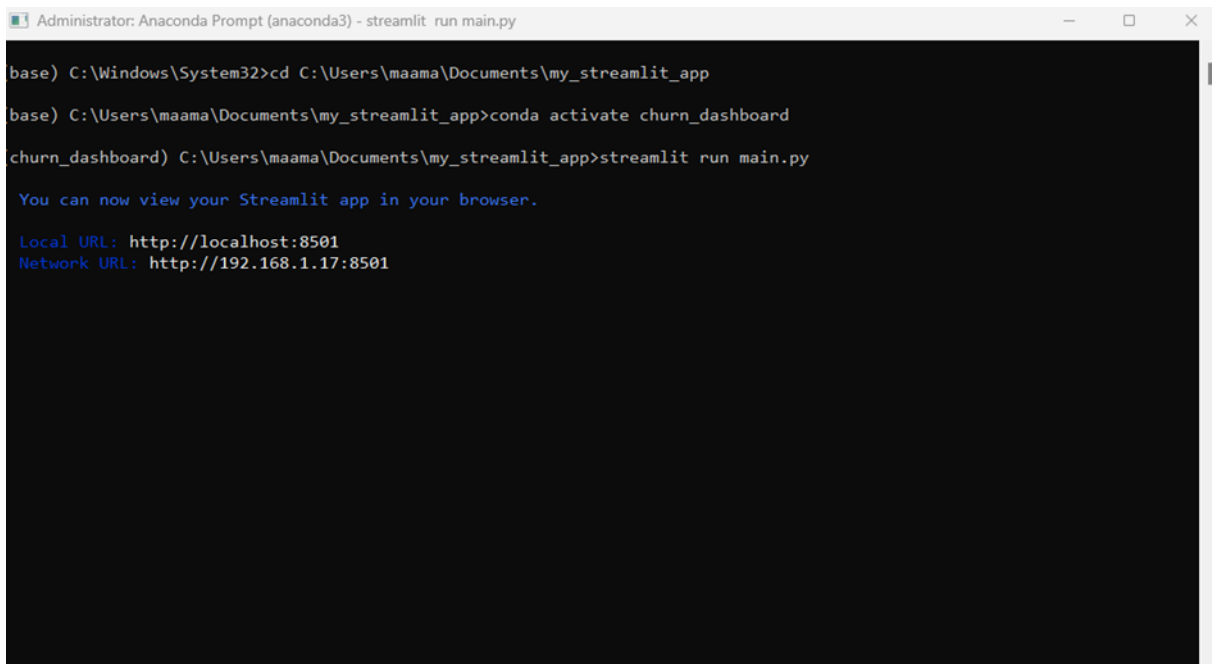
Une intégration en temps réel de ce modèle permettrait de déployer des stratégies proactives pour réduire le churn. Les réseaux LSTM représentent une piste de développement intéressante si davantage de données temporelles sont collectées.

7. Déploiement

Introduction

Ce chapitre présente le déploiement de l'application de prédiction du churn réalisée avec Streamlit. Il détaille la gestion de l'environnement Conda, l'authentification sécurisée, ainsi que les interfaces de visualisation et de prédiction. L'objectif est d'offrir un accès simple, sûr et efficace aux résultats du modèle pour les utilisateurs.

7.1 Gestion des Environnements Conda



```
Administrator: Anaconda Prompt (anaconda3) - streamlit run main.py

base) C:\Windows\System32>cd C:\Users\maama\Documents\my_streamlit_app

base) C:\Users\maama\Documents\my_streamlit_app>conda activate churn_dashboard

churn_dashboard) C:\Users\maama\Documents\my_streamlit_app>streamlit run main.py

You can now view your Streamlit app in your browser.

Local URL: http://localhost:8501
Network URL: http://192.168.1.17:8501
```

Cette capture d'écran montre les étapes clés pour lancer l'application Streamlit depuis l'environnement de développement :

1. Navigation vers le répertoire du projet (`my_streamlit_app`).
2. Activation de l'environnement Conda dédié (`churn_dashboard`).
3. Exécution réussie de l'application via la commande `streamlit run main.py`.

7.1.1 Gestion des requirements

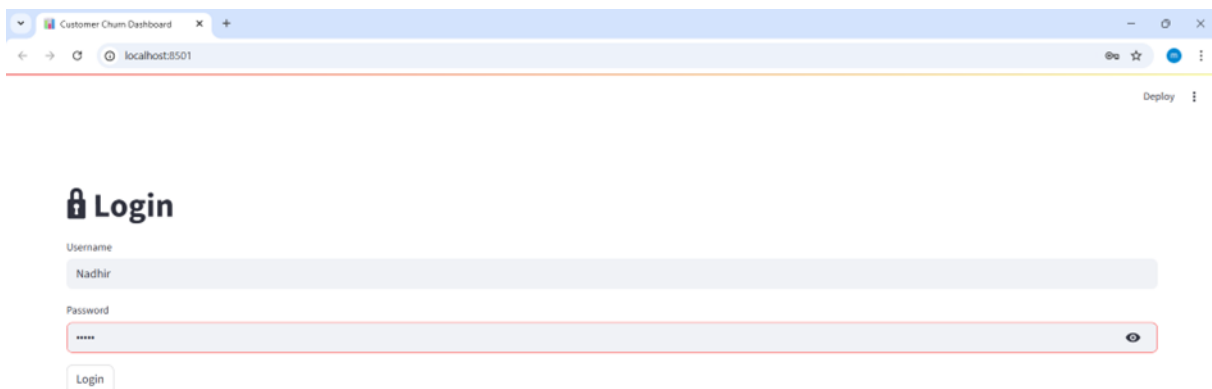
L'utilisation de `conda activate churn_dashboard` confirme la bonne isolation des dépendances du projet, essentielle pour la reproductibilité. Cette étape valide que les bibliothèques sont correctement installées dans l'environnement virtuel.

- python==3.9
- pip:
- streamlit==1.29.0
- pandas==2.1.3
- numpy==1.26.1
- joblib==1.3.2
- matplotlib==3.8.0
- seaborn==0.13.0
- plotly==5.18.0
- scikit-learn==1.3.2
- xgboost==2.0.2
- openpyxl==3.1.2
- Pillow==10.0.1

7.1.2 Lancement Réussi de l'Application

Les URLs générées (<http://localhost:8501> et <http://192.168.1.17:8591>) indiquent que le serveur Streamlit est opérationnel.

7.2 Authentification au dashboard



Cette capture d'écran présente l'interface d'authentification sécurisée de l'application Streamlit dédiée à la prédiction du churn. L'utilisateur "Nadhir" (masqué ici pour des

raisons de sécurité) est en train de saisir ses identifiants pour accéder au tableau de bord.

7.2.1 Sécurité des Données

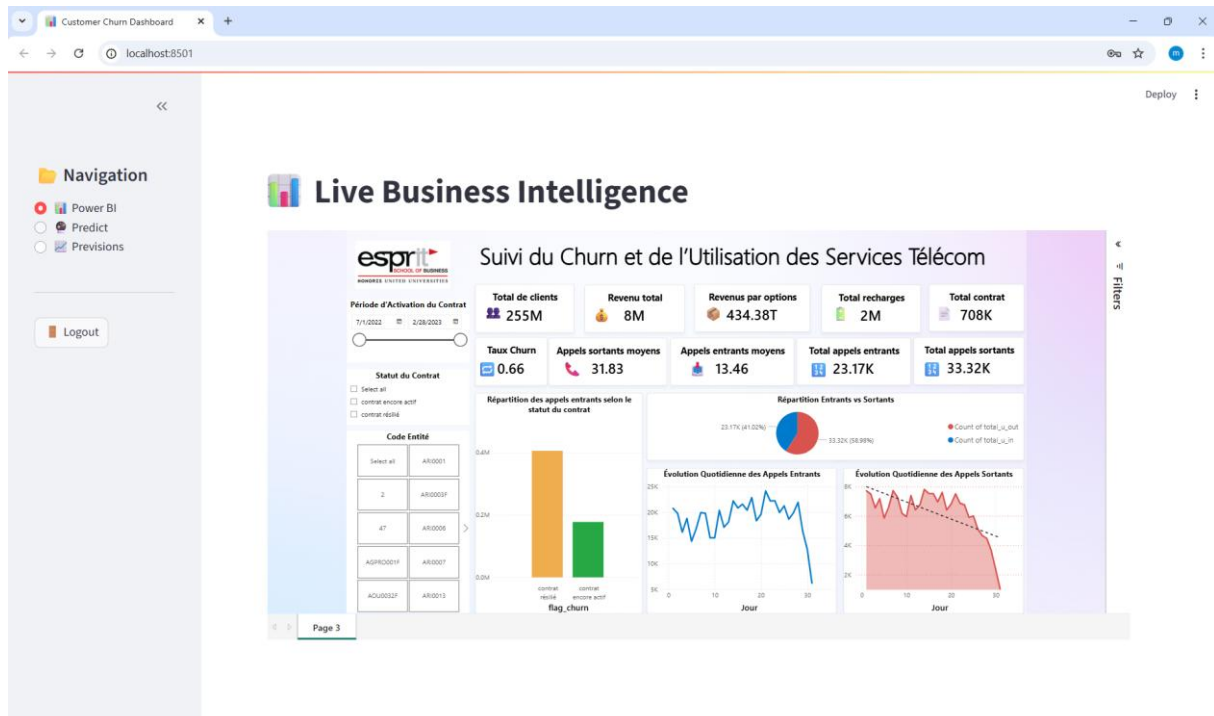
- Le champ *Password* est masqué (affiché en *****), conformément aux bonnes pratiques de sécurité pour protéger les informations sensibles.
- *Implication* : Cela montre une attention aux normes de confidentialité, cruciale pour une application manipulant des données clients.

7.2.2 Personnalisation Utilisateur

- l'application prend en charge une authentification multi-utilisateurs, idéale pour une utilisation en équipe.
- *Fonctionnalité potentielle* : Cela pourrait permettre un accès différencié (ex: droits admin vs. utilisateur standard) en fonction des rôles.

7.3 Power BI

- Le design minimaliste (champs Username, Password, bouton Login) facilite l'expérience utilisateur tout en restant fonctionnel.



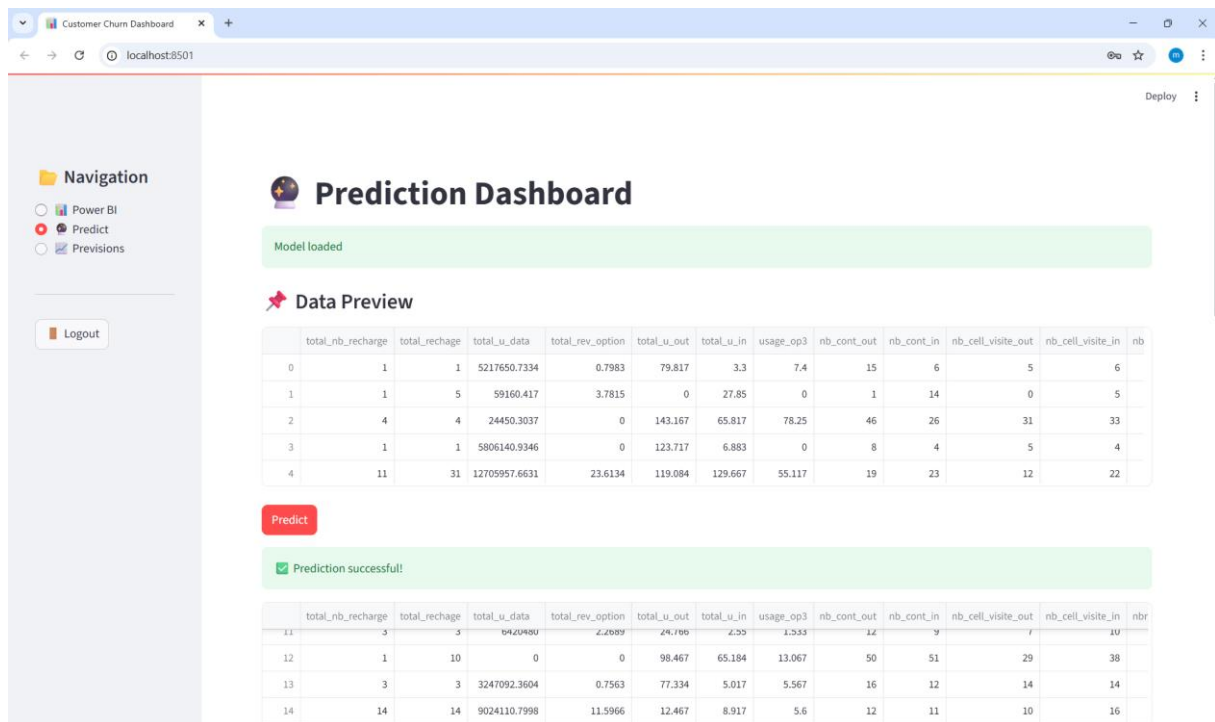
Cette capture d'écran présente un tableau de bord interactif dédié au suivi du churn client et des indicateurs clés du secteur télécom. L'interface est organisée en plusieurs sections :

1. KPI principaux (Taux de churn, appels entrants/sortants, adoption des contrats).
2. Visualisations (répartition des appels par statut de contrat, tendances quotidiennes).

Analyse des Éléments Clés

1. Indicateurs de Performance (KPI)
2. Visualisations et Segmentation
3. Filtres Interactifs

7.4 Prédiction

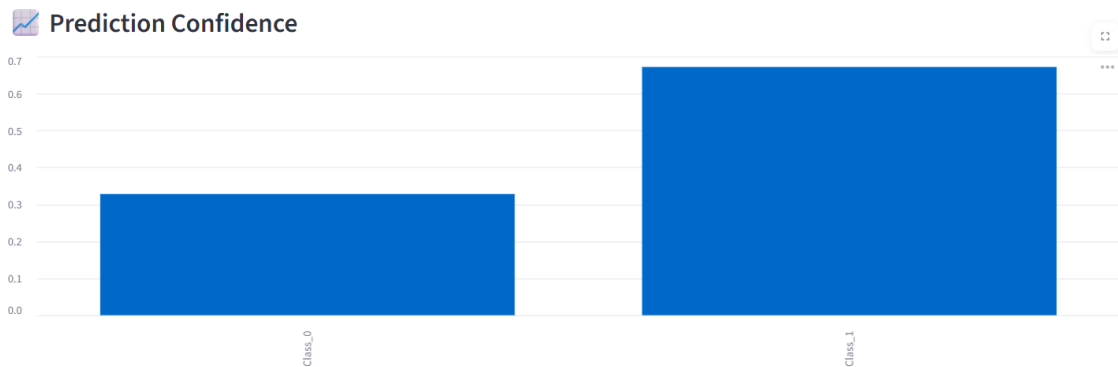
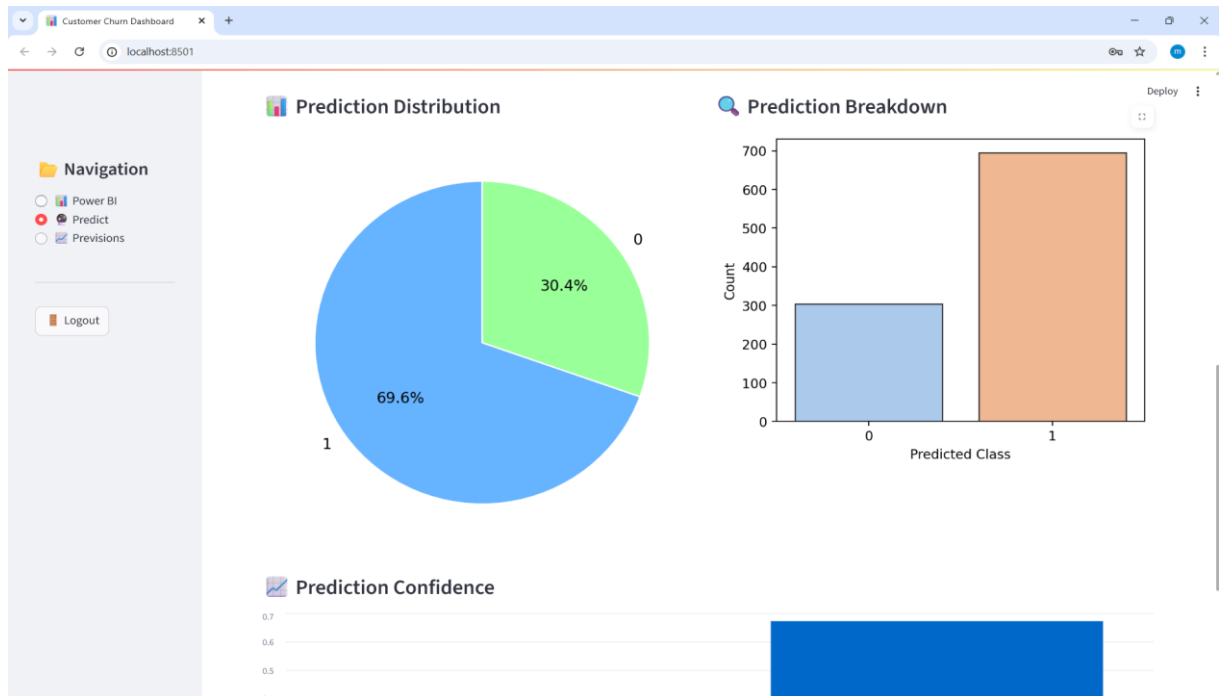


Cette capture d'écran présente l'interface de prédiction du churn depuis l'application

Streamlit. Elle affiche :

1. Un aperçu des données clients utilisées pour la prédiction (variables clés comme `total_u_data`, `total_recharge`, etc.).
2. Le résultat de la prédiction (Prediction successful) avec les valeurs des features pour chaque client analysé confirme le bon fonctionnement du modèle (Random Forest ou autre).

Cette capture d'écran complète la section de prédiction en affichant :



1. La répartition des classes prédites churn sous forme de pourcentages et de comptages.
2. La confiance du modèle pour chaque prédiction, matérialisée par des indicateurs visuels (barres ou scores).

Analyse des Éléments Clés

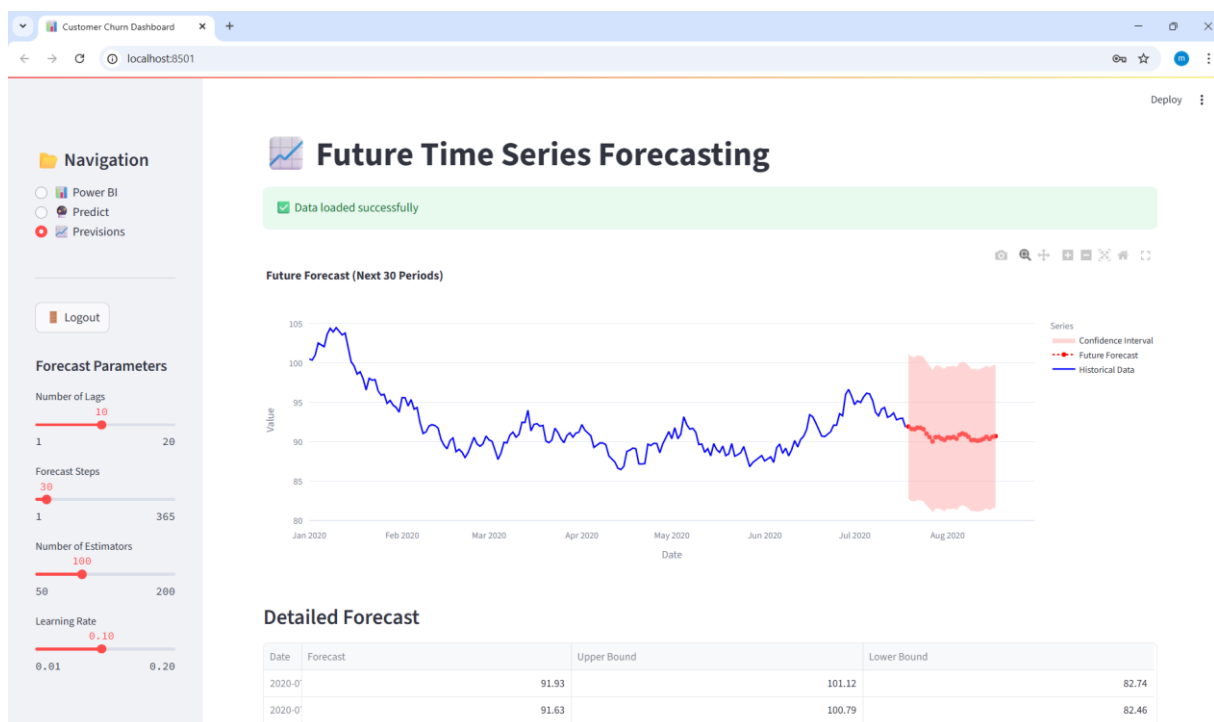
1. Répartition des Prédictions
 - Les clients sont prédits comme fidèles (classe 0), ainsi qu'ils sont identifiés comme à risque de churn (classe 1).

- *Interprétation* : Ce ratio suggère une base client majoritairement stable, mais avec une proportion significative de risques à cibler.
- **Graphique à barres** : Bien que non visible en détail, l'axe vertical (jusqu'à 500) indique un volume important de prédictions traitées.

2. Confiance du Modèle

- Les scores de confiance varient entre 0.4 et 0.8, reflétant la certitude du modèle pour chaque prédiction.

7.5 Prédiction



Cette capture présente une interface dédiée à la prédiction du churn par analyse de séries temporelles, probablement générée via un modèle XGBoost .

1. Paramètres du Modèle

- Number of Lags, Nombre de pas de temps précédents utilisés pour prédire le futur (ex: 10 jours/mois), Plus de lags = meilleure capture des tendances longues, mais risque de sur-apprentissage.
- Forecast Steps à 365 Nombre de périodes futures à prédire (ex: 30 jours). Un horizon trop long réduit la précision. Un horizon court permet des actions ciblées.
- Number of Estimators 100 50 à 200 Nombre d'arbres (pour XGBoost) ou d'itérations (pour SARIMA). Plus d'estimateurs = modèle plus précis, mais calcul plus lent.

- Learning Rate 0.10 0.01 à 0.20 Taux d'apprentissage (pas de gradient pour XGBoost). Un taux faible = convergence lente mais stable. Un taux élevé = risque de divergence.

2. Périodes de Prévision

- Les mois affichés Jan à Aug

Conclusion

Ce chapitre a permis de démontrer la faisabilité et l'efficacité du déploiement d'un outil décisionnel à base de machine learning dans un contexte métier. Grâce à l'utilisation de Streamlit et à une bonne gestion de l'environnement Conda, l'application offre une expérience utilisateur fluide, tout en respectant les standards de sécurité et de performance. Les interfaces développées permettent non seulement de visualiser les indicateurs clés du churn, mais aussi de lancer des prédictions fiables et des prévisions à court ou long terme.

Ce déploiement constitue ainsi une étape clé de la valorisation du projet, rendant les résultats accessibles aux décideurs et ouvrant la voie à une intégration potentielle dans les processus métiers de l'entreprise.

Conclusion Générale

Ce projet a permis de démontrer l'apport concret de la data science dans le domaine des télécommunications, en particulier pour la prédiction du churn client et l'amélioration de la prise de décision. En suivant les étapes de la méthodologie CRISP-DM, nous avons pu structurer efficacement l'ensemble du processus, depuis la compréhension métier jusqu'au déploiement de tableaux de bord interactifs.

L'analyse des données comportementales et transactionnelles a permis d'identifier des facteurs clés de résiliation, tels que le volume d'usage, les habitudes de recharge ou encore le type d'entité. L'intégration de modèles de prévision a renforcé la capacité à anticiper les départs clients, offrant ainsi des pistes concrètes pour améliorer les stratégies de fidélisation.

Enfin, les tableaux de bord développés offrent une visualisation intuitive et accessible des résultats, facilitant leur exploitation par les équipes métiers. Ce projet pose ainsi les bases d'une démarche proactive orientée données, essentielle pour rester compétitif et centré sur le client.